FFT-BASED ACQUISITION OF GNSS SIGNALS

TODD HUMPHREYS

1. The Basic Idea

GNSS signal acquisition can be thought of as a two-dimensional search in Doppler frequency and in pseudorandom code offset. The brute-force approach to acquisition performs standard in-phase and quadrature signal correlation at each cell in the two-dimensional search space. This technique is the simplest to implement and requires very little memory beyond what is used for signal tracking, but is computationally expensive, especially if one wishes to increase acquisition sensitivity by extending the coherent or noncoherent integration time. A cleverer approach, first introduced in [1], exploits the tremendous computational efficiency of the Fast Fourier Transform (FFT) to search simultaneously all possible code offsets at a particular frequency. The result is a remarkably fast acquisition process.

2. Mathematical Background

Fourier Transform: The Fourier transform of the signal $x(t)$ is given by

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \tag{1}$$

Discrete Fourier Transform (DFT): The DFT is basically the Fourier transform of a sampled signal repeated periodically. Suppose $x(t)$ is periodic with period T_0. Let $x_k = Tx(kT)$ be samples of $x(t)$ weighted by the sampling interval $T = T_0/N_0$. The sequence x_k is N_0-periodic; i.e., $x_k = x_{N_0+k}$. Let $X_r = X(rf_0)$ be samples of the Fourier transform $X(f)$ at intervals defined by the fundamental frequency $f_0 = 1/T_0$. Because it represents a Fourier transform of an an N_0-periodic sampled sequence, X_r is also an N_0-periodic sequence. The quantities x_k and X_r are related by

$$X_r \approx \sum_{k=0}^{N_0-1} x_k e^{-j2\pi \frac{rk}{N_0}} \tag{2}$$

$$x_k \approx \frac{1}{N_0} \sum_{r=0}^{N_0-1} X_r e^{j2\pi \frac{rk}{N_0}} \tag{3}$$

Date: October 21, 2024.

The approximations become exact as $T \to 0$. Equations (2) and (3) (with $\approx$ replaced by $=$) define the direct and inverse DFTs. The notation

$$x_k \Longleftrightarrow X_r \tag{4}$$

is used to indicate that x_k and X_r are a DFT pair.

Conjugate Symmetry Property: If the sequence x_k is real with DFT X_r, then

$$X_{-r} = X_r^* \tag{5}$$

This property follows directly from the definition of the DFT. Note, however, that in general $X_{-r} \neq X_r^*$ for a complex sequence x_k.

Time Shifting Property: If $x_k \Longleftrightarrow X_r$, then

$$x_{k-n} = X_r e^{-j2\pi \frac{nr}{N_0}} \tag{6}$$

Circular Convolution: For two N_0-periodic sequences f_k and g_k, circular convolution is defined by

$$f_k \circledast g_k = \sum_{n=0}^{N_0-1} f_n g_{k-n} \tag{7}$$

Circular Correlation: This is a close cousin of circular convolution. For two N_0-periodic sequences f_k and g_k, circular correlation is defined by

$$\mathrm{Corr}(f,g)_k = \sum_{n=0}^{N_0-1} f_n g_{k+n} \tag{8}$$

Discrete Convolution Theorem: An important property of circular convolution is that if $f_k \Longleftrightarrow F_r$ and $g_k \Longleftrightarrow G_r$, then

$$f_k \circledast g_k \Longleftrightarrow F_r G_r \tag{9}$$

In other words, convolution in the time domain amounts to multiplication in the frequency domain.

Discrete Correlation Theorem: The close connection between circular correlation and circular convolution suggests that correlation should have a theorem too. This is indeed the case. If f_k and g_k are real sequences, and if $f_k \Longleftrightarrow F_r$ and $g_k \Longleftrightarrow G_r$, then

$$\mathrm{Corr}(f,g)_k \Longleftrightarrow F_r^* G_r \tag{10}$$

In other words, for real sequences, correlation in the time domain amounts to complex conjugate multiplication in the frequency domain. Because this theorem is central to the FFT-based acquisition technique, it is worthwhile to see how it is derived. Let f_k and g_k be (possibly complex) discrete N_0-periodic sequences with respective DFTs F_r and G_r, and let the circular correlation of these sequences be given by

$$z_k = \mathrm{Corr}(f,g)_k = \sum_{n=0}^{N_0-1} f_n g_{k+n} \tag{11}$$

Now consider the DFT of z_k:

$$Z_r = \sum_{k=0}^{N_0-1} z_k e^{-j2\pi \frac{rk}{N_0}} \tag{12}$$

$$= \sum_{k=0}^{N_0-1} \sum_{n=0}^{N_0-1} f_n g_{k+n} e^{-j2\pi \frac{rk}{N_0}} \tag{13}$$

Changing the order of summation yields

$$Z_r = \sum_{n=0}^{N_0-1} f_n \sum_{k=0}^{N_0-1} g_{k+n} e^{-j2\pi \frac{rk}{N_0}} \tag{14}$$

The second summation in Eq. (14) is simply the DFT of g_{k+n}. By the time shifting property,

$$g_{k+n} \Longleftrightarrow G_r e^{j2\pi \frac{nr}{N_0}} \tag{15}$$

which leads to

$$Z_r = \sum_{n=0}^{N_0-1} f_n G_r e^{j2\pi \frac{nr}{N_0}} \tag{16}$$

$$= G_r \sum_{n=0}^{N_0-1} f_n e^{j2\pi \frac{nr}{N_0}} \tag{17}$$

$$= G_r F_{-r} \tag{18}$$

In the special case that f_k is a real sequence, then by the conjugate symmetry property $F_{-r} = F_r^*$, yielding

$$z_k \Longleftrightarrow Z_r = F_r^* G_r \tag{19}$$

which completes the derivation. Note that the derivation of Eq. (19) only actually requires that f_k be a real sequence; g_k is allowed to be complex.

3. Implementation

Supported by the above mathematical theory, one can lay out a step-by-step approach to FFT-based GNSS acquisition. Let x_k be a length-N_0 signal sample sequence output by the GNSS front-end; x_k could be a complex baseband signal or a real-valued signal centered at some intermediate frequency. The sequence x_k represents a superposition of signals from all the visible GNSS SVs in the passband of the front-end, plus thermal noise. Suppose one would like to acquire a particular signal embedded in x_k—the target signal. The target signal's spreading code will be assumed to be N_0-periodic (there are N_0 front end samples in the target signal' spreading code). In truth, it will not be exactly N_0-periodic because there are usually not an integer number of front-end samples per code, but the difference is insignificant. The rest of the GNSS signals present in x_k look like noise from the perspective of the target signal. Let c_k be the length-N_0 sampled local replica of the spreading code corresponding to the target signal, with sampling interval T. The following table describes the FFT-based acquisition algorithm.

TABLE 1. FFT-based Acquisition Algorithm

(1) Take the DFT of c_k to get C_r.
(2) Choose a test frequency f_i. If x_k is a zero-IF complex baseband signal, then f_i will be equal to the test Doppler frequency $f_{D,i}$; if x_k is a real bandpass signal with intermediate frequency f_{IF}, then $f_i = f_{D,i} + f_{IF}$.
(3) Perform complex mixing by multiplying the sequence x_k by $\exp(-j2\pi f_i t_k)$, where $t_k = kT$, $k = 0, 1, ..., N_0 - 1$. Call the resulting complex sequence $\tilde{x}_k$. Complex mixing is a fancy way of saying that x_k is mixed by both the in-phase and the quadrature local carrier replicas, which becomes obvious when one expands $\exp(-j2\pi f_i t_k)$ as
$$\exp(-j2\pi f_i t_k) = \cos(2\pi f_i t_k) - j\sin(2\pi f_i t_k)$$
(4) Take the DFT of $\tilde{x}_k$ to get $\tilde{X}_r$.
(5) Multiply $\tilde{X}_r$ by the complex conjugate of C_r; i.e., calculate $Z_r = \tilde{X}_r C_r^*$.
(6) Take the inverse DFT of Z_r to obtain the correlation sequence z_k.
(7) Find the index k that maximizes $|z_k|^2$. Call this index k_{max}. If $|z_{k_{max}}|^2$ exceeds a pre-determined signal detection threshold, then stop: k_{max} can be assumed to correspond to the correct code offset. If $|z_k|^2$ is below the detection threshold for all values of k, then increment i. If f_i is now beyond the frequency search window, then quit: the target signal is not present or is not strong enough to be detected with only one code accumulation. If f_i is still within the frequency search window, then go to step (3).

4. Further Considerations

4.1. **FFT Implementation.** Clearly, one would like to implement the DFT as an FFT to speed computations. This is obvious when one considers that the standard DFT's computational cost is on the order of N_0^2 operations, whereas the FFT's cost is on the order of $N_0 \log_2(N_0)$ operations (for N_0 a power of 2) or $N_0 \log_4(N_0)$ operations (for N_0 a power of 4). Of course, one should not expect to be so lucky as to have an RF front-end that produces exactly a power-of-two number of samples per code period. Two alternatives are available for forcing N_0 to be a power of two.

The first option is to undersample or oversample the incoming sequence x_k. Oversampling is usually the best option since the native sampling rate $1/T$ is probably not much wider than the GNSS signal's Nyquist bandwidth, and therefore undersamping would risk introducing aliasing effects. One oversamples x_k by linear interpolation to generate a power-of-two (or four) number of samples per code. The oversampling operation can be coded very efficiently.

Another approach is to take in N_0 front-end samples for acquisition, where N_0 is a power of two and is larger than the number of front-end samples in two consecutive codes. The resulting sequence x_k is now no longer N_0-periodic, but it contains at least one full code. For correlation, the local code replica sequence c_k is zero-padded out to N_0 elements. At some offset, the code

replica in this zero-padded sequence will align with the full code in x_k and produce a strong correlation peak. Therefore, the step-by-step process outlined above can be applied directly to the elongated x_k and the zero-padded c_k sequences. This approach can be more computationally expensive than the oversampling approach—depending on the target platform—but it avoids the problem of data bit transitions, discussed next.

4.2. **The Effect of Data Bit Transitions.** If one adopts the oversampling approach to ensure that N_0 is a power of two, then one must be aware that the oversampled data interval spanning one code period may contain a navigation data bit transition. In fact, the only way to avoid periodic bit transitions is to perfectly align the data capture window with the code of the signal for which one wishes to search. But the whole purpose of the code offset search is to find the offset that would make such an alignment possible! Instead, one simply accepts the fact that every so often a captured front-end segment (one code period in length) will be corrupted by a data bit transition. The effects are not all that serious. For example, data bit transitions occur on average only once in 40 GPS L1 C/A code periods, and, even when they occur, they may not completely annihilate the correlation peak because, in order to do so, they must occur exactly halfway into the captured data interval. This property is used to advantage in data bit synchronization after acquisition. Note that because the zero-padding approach performs correlation with a complete code, it does not suffer from data bit transition effects.

4.3. **Improving Acquisition Sensitivity.** Until now, only acquisition using via correlation over one code period has been considered. As it turns out, this is adequate for reliable signal detection of incoming signals with carrier-to-noise ratios (C/N_0) greater than about 43 dB-Hz. To reliably acquire signals with $C/N_0 < 43$ dB-Hz, one can either (1) perform coherent correlation over intervals longer than one code, or (2) add up the coherent correlation products of several short correlation intervals (noncoherent correlation), or (3) do a combination of these. The DSP implementation of FFT-based acquisition, discussed below, considers these approaches to improving sensitivity.

5. FFT-based Acquisition Implementation on the UT/Cornell GRID Receiver

To make the foregoing discussion on FFT-based acquisition more concrete, consider acquisition on the UT/Cornell GRID receiver (GRID stands for *General Radionavigation Interfusion Device*). An embedded version of GRID is implemented on a Texas Instruments C6400 series DSP. Like other DSP manufacturers, Texas Instruments provides a library of FFT routines that have been optimized for use on the DSP [2]. For computational efficiency, these routines are uninterruptible, they require the number of points N in the FFT to be a power of 2, and they require N 16-bit complex factors to be computed beforehand and stored in (preferably on-chip) memory. The following FFT-based acquisition strategy has been designed around these restrictive but efficient FFT library functions.

5.1. **Cost/Sensitivity Analysis.** Acquisition sensitivity is increased by increasing the coherent or noncoherent averaging time in each search cell. Increasing the coherent averaging time

T_c is more effective than increasing the number K of noncoherent averages (as measured by the length of the data interval required to achieve a given acquisition threshold), but, for several reasons, increasing T_c beyond a certain limit becomes computationally impractical.

Consider the following cost/sensitivity analysis for FFT-based acquisition on the 'C6416 using the DSP Library function `fft16x32()`. It can be shown that the total computation time required for acquisition on one channel with $F_s = 5.714$ MHz, $T_c = 1$ ms, $N = 8192$, N_f Doppler search frequencies, and K noncoherent integrations (with all data stored on chip) is given by

$$T_{acq} = 182 + 310 + 415N_f + KN_f(204 + 310 + 310)$$

where time is expressed in μs. In the above equation, the quantities 182, 415, and 204 scale linearly with N, whereas the three 310s, which represent FFT calculations, scale as[2]

$$T_F = C_t \left[\left(\frac{13N}{8} + 24 \right) \text{ceil}\left[\log_4(N) - 1\right] + \frac{3(N+8)}{2} + 27 \right]$$

with $C_t = 3.36 \times 10^{-3}$. As T_c increases, N increases to the minimum power of 2 that can accommodate $N_s = TF_s$ front-end samples (set aside for now issues of limited memory and limitations on the maximum value of N). Also, as T_c increases, N_f must increase by the same factor to avoid loss of signal power over the coherent integration interval. Finally, as T_c increases, the probability of encountering a navigation data bit transition increases. To approximate this event as an equivalent computational penalty, assume that each coherent integration interval that straddles a navigation bit start/stop time must be discarded and replaced by another coherent integration interval of equal length. Thus for $T_c = 1$ ms and $K = 20$, a total of 21 1-ms coherent intervals must be computed, and for $T_c = 10$ms and $K = 2$, a total of 3 10-ms coherent intervals must be computed.

To complete the cost/sensitivity analysis, assume that the number K of noncoherent integrations is chosen at each carrier-to-noise (C/N_0) ratio to be the minimum K that guarantees "reliable" acquisition, where "reliable" is interpreted as a probability of detection $P_D \geq 0.95$ and a probability of false alarm $P_{FA} = 0.01$. (Note that P_{FA} is the probability that *any* cell in the $N_f \times N$ search space exceeds the acquisition threshold. For an introduction to the hypothesis testing methods used to calculate P_{FA} and P_D, see [3], Appendix B. Note that there is a sign error in the argument of the exponential function in [3]'s Eq. (B7): the term $-2K\beta$ should be $+2K\beta$.)

The above analysis can be combined with an assumed Doppler resolution of $N_f/T_c = 35$ bins per ms to generate the plot shown in Fig. 1, which gives the computation time as a function of C/N_0 for several different values of T_c. (Note that computation time as expressed in Fig. 1 should not be confused with the length of the front-end data interval required for acquisition: the latter is much less than the former.) Figure 1 indicates that an acquisition strategy based on $T_c = 1$ ms is computationally more efficient than strategies based on longer coherent integration times for values of C/N_0 down to 24 dB-Hz. Adding to this result the facts that a longer T_c requires more memory and that the DSP Library function `fft16x32()` limits N to 2^{15}, the choice of acquisition strategy becomes clear: GPS L1 C/A-code acquisition in the DSP should be based on noncoherent sums of $T_c = 1$-ms coherent integration intervals.

Figure 1 is also useful for choosing a practical acquisition threshold. In the current implementation, a threshold of $C/N_0 = 32$ dB-Hz was chosen to limit the computation time for each PRN to less than 2 seconds.

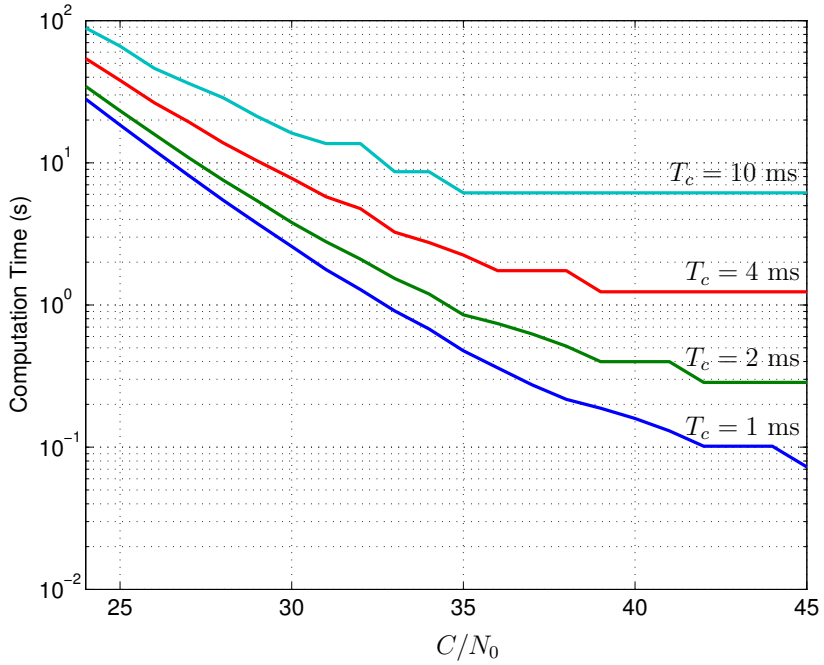

FIGURE 1. Computation time required for reliable acquisition as a function of C/N_0 for several values of the coherent integration interval T_c.

5.2. **Implementation.** The implementation of FFT-based acquisition in the DSP is illustrated in block diagram form in Fig. 2. Before circular correlation, the front-end sign and magnitude bits must be converted to integers for use in the FFT and resampled using linear interpolation to arrive at an even power of 2. These operations represent an inefficiency relative to the bit-wise correlation strategy, which operates directly on the 5714 sign and magnitude bits, but the DSP handles the conversion and resampling quickly (24 μs to convert 5174 front-end samples and 52 μs to interpolate these into 8192 samples), and the inefficiency is more than compensated by the efficiency of the FFT-based circular correlation. As an alternative to resampling, one might consider zero-padding, but this requires $N = 16384$ and increases memory use.

The results of the circular correlation operation are squared, summed, and accumulated non-coherently K times at each Doppler search frequency. If the maximum value of the resulting statistic across all frequencies and code offsets exceeds a predetermined threshold, then a successful acquisition is declared and the corresponding Doppler frequency and code offset are sent to the tracking loops. During real-time operation with a CPU laboring to track several channels, this "maximum value" approach can yield Doppler frequency and code phase estimates that are up to 2 seconds older than the incoming data. This latency does not obsolete the Doppler frequency estimate, which varies little over 2 seconds, but it does require the code

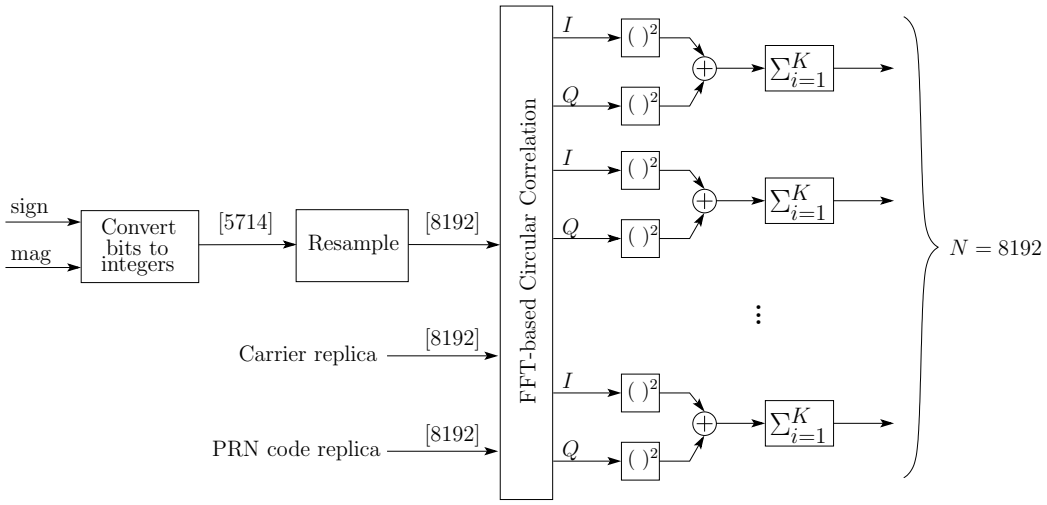

FIGURE 2. Block diagram of the FFT-based acquisition implementation in the DSP

phase estimate to be propagated forward to the epoch of the current incoming data. This propagation makes use of the following relation:

$$N_s = \frac{N_{s,0}}{1 + \frac{s_m f_D}{f_{L1}}}$$

where N_s is the value for the number of samples per C/A code period that is used for propagation, $N_{s,0}$ is the nominal number of samples per C/A code period assuming zero Doppler shift and a perfect receiver clock, $s_m = -1$ (1) for high (low)-side mixing in the RF front end, f_D is the apparent Doppler shift as measured by acquisition, and $f_{L1} = 1575.42$ MHz is the L_1 carrier frequency. As an alternative to the "maximum value" approach, one might terminate the search process as soon an any cell exceeds the predetermined threshold, thus eliminating the need for propagation. This method, however, suffers from a higher incidence of false alarm because of minor correlation peaks in the code offset dimension and side lobes in the frequency dimension. Hence, the "maximum value" approach is preferred. For a maximum frequency error of $\Delta f = 175$ Hz, the above propagation relation is accurate to 0.23 C/A code chips over a propagation interval of 2 seconds.

The acquisition search strategy implemented in the DSP begins by performing a quick search for strong signals ($C/N_0 \geq 42$ dB-Hz) using $K = 2$. Thereafter, K is gradually increased to 41. In the steady-state, the search strategy allocates 65% of its time to deep searches ($K = 41$) and 35% of its time to shallow searches ($K = 2$). The total on-chip memory footprint for the C/A-code FFT-based acquisition is 240 kB.

REFERENCES

[1] D. Van Nee and A. Coenen, "New fast GPS code-acquisition technique using FFT," *Electronics Letters*, vol. 27, no. 2, pp. 158–160, 1991.

[2] Texas Instruments, *TMS320C64x DSP Library Programmer Reference (SPRU565B)*, Oct. 2003. www.ti.com.

[3] A. J. Van Dierendonck, *Global Positioning System: Theory and Applications*, ch. 8: GPS Receivers, pp. 329–407. Washington, D.C.: American Institute of Aeronautics and Astronautics, 1996.

The University of Texas at Austin

Email address: todd.humphreys@utexas.edu