FFT-Based Acquisition of GPS L2 Civilian CM and CL Signals

by Mark L. Psiaki
Cornell University, Ithaca, N.Y.

BIOGRAPHY

Mark L. Psiaki is an Associate Professor of Mechanical and Aerospace Engineering at Cornell University. He received a B.A. in Physics and M.A. and Ph.D. degrees in Mechanical and Aerospace Engineering from Princeton University. His research interests are in the areas of estimation and filtering, spacecraft attitude and orbit determination, and GNSS technology and applications.

ABSTRACT

Block processing Fast Fourier Transform (FFT) algorithms have been developed to do signal acquisition for the new CM and CL civilian codes that will appear on the L2 GPS frequency once the Block IIR-M satellites are launched. These algorithms have been developed to aid in the acquisition of weak GPS signals. FFT block processing can be used to significantly reduce the computational burden caused by the extended lengths of the new codes and by the long integration times needed to acquire weak signals. The target application for this technology is high-altitude spacecraft navigation. The new algorithms must work efficiently when the maximum allowable FFT batch size precludes the use of a single FFT for a full code period. Zero-padding and overlap-and-discard techniques are used to generate partial correlation accumulations at a range of pseudo-random number (PRN) code start times, and interpolation techniques are used to map these partial accumulations onto a desired grid of start times before summation into full accumulations. Frequency-domain techniques are used to reduce the number of FFT and inverse FFT (IFFT) operations that are needed if one performs the acquisition search over a range of frequencies. The new algorithms have been used to acquire simulated signals with carrier-to-noise ratios as low as 9 dB-Hz. They can reduce execution times in comparison to brute-force algorithms by factors ranging from 1,000 to over 30,000.

INTRODUCTION

A GPS receiver acquires a signal by determining its PRN code start time and carrier Doppler shift. It does this by searching for the maximum detection statistic over the full range of possible code start times and Doppler shifts. At each code start time and Doppler shift the detection statistic is calculated using baseband mixing, code correlation, coherent integration, squaring, and non-coherent integration.

The present paper develops efficient ways to calculate detection statistics at multiple code start times and Doppler shifts when acquiring the new civil-moderate (CM) and civil-long (CL) codes at low received power levels. These codes will begin appearing on the L2 frequency when the first Block IIR-M satellites are launched [1]. Weak CL and CM signals are easier to acquire and track than weak L1 C/A code signals because they have lower cross-correlation between different codes and because the CL code does not carry navigation data. Acquisition of weak signals requires the use of long coherent and non-coherent integration intervals. The required number of calculations grows with the length of these integration intervals, which is why it is important to use efficient acquisition calculations.

There are several reasons for wanting to acquire very weak GPS signals. High-altitude spacecraft navigation, i.e., navigation at altitudes above the GPS constellation, can benefit from an ability to use weak side-lobe signals [2,3]. Use of side-lobe signals can significantly increase the number of available satellites at very high altitudes, but most importantly can reduce or eliminate periods when no GPS signals are available. The improved signal observability will improve the robustness of the navigation solution and may alleviate the requirement for a very stable reference oscillator in the receiver. Another application that would benefit from the ability to acquire weak signals is E911 service. The current work has been motivated by the high-altitude spacecraft navigation application, and it considers acquisition scenarios that are typical of high-altitude orbits.

This work builds on a number of previous results. It uses the standard acquisition approach that is described in Ref. 4, but it uses FFT-based block processing methods like those described in Ref. 5 in order to expedite the

calculation of multiple acquisition statistics. Reference 6 extends the methods of Ref. 5 to use long coherent and non-coherent integration intervals for the acquisition of weak L1 C/A signals. The present work adapts the methods of Ref. 6 to the L2 CM and CL signals. The most significant change is to introduce the use of zero padding from Ref. 7 in the FFT calculations of Ref. 6. These techniques enable CL code acquisition using FFT blocks that are shorter than the CL code period. It is impractical to use block lengths that equal the CL code length due to memory limitations. In the case of the CM code, zero-padded FFT block processing is used to slide the coherent integration intervals with respect to the incoming data stream as the code start time varies. This approach ensures that the correct code start time produces coherent integration intervals which do not cross navigation data symbol transitions.

This paper makes several contributions to the art of GPS signal acquisition via FFT block processing. First, it presents a simple means of dealing with the time-multiplexed nature of the CM and CL codes. Second, it adapts the zero-padding methods described in Ref. 7 to work over long data spans that can include multiple code periods. The principal novelty of this approach is the use of interpolation of correlations onto a grid of code start times. Interpolation is required when the code Doppler shift used in the FFT calculations does not match that of the received signal closely enough, or when the code period does not equal an integer number of RF sample periods. Third, it adapts the frequency-shifting and coarse-frequency-grid/fine-frequency-grid approaches of Ref. 6 to the situation of zero-padding and long codes. These methods help reduce the needed number of computations when calculating acquisition statistics at more than one Doppler shift. Fourth, it shows how to include an a priori estimate of the rate of change of the Doppler shift (i.e., a carrier phase acceleration) in the calculations. Aiding by an a priori acceleration estimate is needed in order to keep the power from leaking between different frequency bins when working with long coherent integration intervals. Fifth, it analyzes the operations counts and memory requirements of the new algorithms. Sixth, it tests these techniques under weak signal conditions using the new civilian L2 CL and CM codes.

This paper's algorithms and results are presented in 6 main sections. A model of the L2 civilian signal is presented in Section II along with a review of the signal acquisition calculations. Section III develops the FFT-based block processing algorithms for calculating the acquisition statistic at multiple code start times for a given initial Doppler shift and a given Doppler shift rate. Section IV shows how circular frequency shifting allows one to re-use many of the FFT calculations of Section III in order to calculate acquisition statistics for a set of alternate Doppler shifts. Section V adapts the coarse-frequency-grid/fine-frequency-grid approach of Ref. 6 for use with zero-padded calculations. This approach further reduces the required number of FFT and IFFT calculations when searching over multiple Doppler shifts if a single coherent integration interval is broken up into multiple FFT blocks. Section VI analyzes the operations counts and memory requirements of the CL and CM acquisition algorithms. Section VII presents acquisition test results using data from hardware and software simulators. Section VIII summarizes the paper's results and conclusions.

II. SIGNAL MODEL AND REVIEW OF SIGNAL ACQUISITION CALCULATIONS

The acquisition calculations are based on the following model of the signal as it comes out of an RF front end:

$$
\begin{aligned}
y_k = A \{ & d[(1+h_0)(kDt-t_s) + 0.5x(kDt-t_s)^2] \\
& \times c_{M0}[(1+h_0)(kDt-t_s) + 0.5x(kDt-t_s)^2] \\
& + c_{0L}[(1+h_0)(kDt-t_s) + 0.5x(kDt-t_s)^2]\} \\
& \times cos[w_{IF}kDt + f_0 + w_{D0}kDt + 0.5a_D(kDt)^2] \\
& + n_k
\end{aligned}
\tag{1}
$$

The quantities in this model are the RF front end output y_k at sample time kDt, the sample period Dt, the carrier amplitude A, the GPS navigation data symbol stream $d[t]$, the L2 civilian CM PRN code time history interspersed with zeros $c_{M0}[t]$, the L2 civilian CL PRN code time history interspersed with zeros $c_{0L}[t]$, the start time of the CM and CL PRN codes t_s, the initial fractional code Doppler shift h_0, the fractional code Doppler shift rate x, the nominal intermediate value of the L2 carrier frequency w_{IF}, the initial carrier phase f_0, the initial carrier Doppler shift w_{D0}, the carrier Doppler shift rate a_D, and the thermal/interference noise n_k. The value of w_{IF} is the result of mixing and intentional in the RF front end.

Three of the signals in eq. (1) represent bit or chip values. The navigation data symbol stream $d[t]$ is a sequence of +1 and -1 values that switch randomly every 0.020 sec. The $c_{M0}[t]$ PRN code is a known pseudo-random sequence of +1/0/-1 chip values that chip at a nominal rate of 1.023 MHz. The subscript $()_{M0}$ indicates that every other chip is a +1/-1 value from the L2 civilian CM code with a zero value between each code chip. Similarly, the $c_{0L}[t]$ PRN code is a known +1/0/-1 sequence with a nominal chipping rate of 1.023 MHz that has zeros interspersed every other chip between the +1/-1 values of the L2 civilian CL code. The $c_{M0}[t]$ code starts with the first CM chip followed by a zero followed by the second CM chip, and so on. The $c_{0L}[t]$ code starts with a zero followed by the first CL chip followed by another zero, followed by the second CL chip, and so on. This arrangement models the time multiplexing of the CM and CL codes in a way that facilitates the developments of this paper while preserving the code's autocorrelation and cross-correlation properties. The CM code repeats after 10,230 chips, which makes $c_{M0}[t]$ periodic with a period of 20,460 chips

or 0.020 sec. Each CM code period is aligned with a $d[t]$ navigation data symbol interval. The CL code repeats after 767,250 chips, which gives $c_{0L}[t]$ a repetition period of 1,534,500 chips or 1.5 sec.

The code and carrier Doppler shifts and Doppler shift rates are related if one neglects ionospheric effects. The relationships are: $h_0 = (w_{D0} + a_D t_s)/w_{L2}$ and $x = a_D/w_{L2}$ where $w_{L2} = 2p \times 1227.6 \times 10^6$ rad/sec.

Calculation of the signal acquisition statistic involves mixing with a replica signal, coherent integration, and non-coherent integration [4]. The mixing and coherent integration for CL code acquisition takes the following form:

$$I_{0L(m)}[\hat{t}_s, \hat{w}_{D0}] =$$
$$\sum_{k=mn_{coh}}^{(m+1)n_{coh}-1} y_k c_{0L}[(1+\hat{h}_0)(kDt-\hat{t}_s) + \frac{\hat{x}}{2}(kDt-\hat{t}_s)^2]$$
$$\times cos[w_{IF}kDt + \hat{w}_{D0}kDt + \frac{\hat{a}_D}{2}(kDt)^2] \quad (2a)$$

$$Q_{0L(m)}[\hat{t}_s, \hat{w}_{D0}] =$$
$$\sum_{k=mn_{coh}}^{(m+1)n_{coh}-1} y_k c_{0L}[(1+\hat{h}_0)(kDt-\hat{t}_s) + \frac{\hat{x}}{2}(kDt-\hat{t}_s)^2]$$
$$\times sin[w_{IF}kDt + \hat{w}_{D0}kDt + \frac{\hat{a}_D}{2}(kDt)^2] \quad (2b)$$

The quantities $I_{0L(m)}$ and $Q_{0L(m)}$ are, respectively, the in-phase and quadrature accumulations for the mth coherent accumulation interval. This interval starts with sample $m^\prime n_{coh}$ and sums over n_{coh} samples. The quantities $\hat{t}_s$, $\hat{h}_0$, $\hat{x}$, $\hat{w}_{D0}$, and $\hat{a}_D$ are the acquisition's trial values for, respectively, the CL code start time, the initial code Doppler shift, the code Doppler shift rate, the initial carrier Doppler shift, and the carrier Doppler shift rate. The quantities $I_{0L(m)}$ and $Q_{0L(m)}$ are dependent on the chosen trial values of $\hat{t}_s$ and $\hat{w}_{D0}$. The trial values of $\hat{h}_0$ and $\hat{x}$ are computed from $\hat{w}_{D0}$ and $\hat{a}_D$ using the previously defined relationships between code and carrier Doppler shift and Doppler shift rate. The quantity $\hat{a}_D$ is assumed known as an external aid to the acquisition calculations. The mixing and coherent integrations for the CM PRN code are similar, but they replace $c_{0L}[t]$ with $c_{M0}[t]$, and they use a summation range that varies as a function of $\hat{t}_s$ in order to avoid summing over a navigation data symbol transition.

The coherent integration calculations are followed by a non-coherent summation. The resulting acquisition statistic for the CL code is

$$P_{0L}[\hat{t}_s, \hat{w}_{D0}] =$$
$$\sum_{m=0}^{M-1} \{I_{0L(m)}^2[\hat{t}_s, \hat{w}_{D0}] + Q_{0L(m)}^2[\hat{t}_s, \hat{w}_{D0}]\} \quad (3)$$

where M is the number of coherent integration intervals whose correlation powers are summed to compute the statistic. The CM acquisition uses a similar non-coherent summation formula.

An acquisition is carried out by computing the statistic in eq. (3) on a 2-dimensional $[\hat{t}_s, \hat{w}_{D0}]$ grid that covers the entire range of possible values. If there are n_{ts} points in the $\hat{t}_s$ grid direction and n_{fine} points in the $\hat{w}_{D0}$ grid direction, then acquisition using a brute-force application of eqs. (2a)-(3) requires $n_{fine}M[n_{ts}(4 + 10n_{coh}) + 7n_{coh}]$ floating point operations. This can be a huge number when trying to detect weak signals because n_{coh} may be as large as 100,000 for the CM code and as large as 5,000,000 or more for the CL code while n_{ts} must be 20,460 or a multiple thereof for the CM code and 1,534,500 or a multiple thereof for the CL code. The n_{coh} ranges are determined by the lengths of the coherent integration intervals; long intervals are needed in order to efficiently detect weak signals. The n_{ts} ranges result from the requirement to search over all possible start times of the periodic PRN codes at a resolution of one chip or better. Section III explains how FFT block processing can significantly reduce the $n_{ts}n_{coh}$ term in this operation count. Sections IV and V describe methods for reducing the coefficient of n_{fine} in this count; that is, they explain how to search efficiently along the frequency direction.

III. FFT BLOCK PROCESSING TECHNIQUES FOR SEARCHING OVER CL AND CM CODE START TIMES

A. CL Code Start Time Acquisition Calculations

The FFT and IFFT functions can be used to reduce the amount of computation needed to evaluate eqs. (2a) and (2b) in cases where accumulations are calculated at multiple code start times $\hat{t}_s$. This process starts by breaking the coherent summations of eqs. (2a) and (2b) into shorter partial coherent integrations. This is often necessary in order to limit the length of the FFTs that get used in the computation. The partial coherent integrations take the forms:

$$DI_{0L(m,l)}[\hat{t}_s, \hat{w}_{D0}] =$$
$$\sum_{k=mn_{coh}+Ll}^{mn_{coh}+L(l+1)-1} y_k c_{0L}[(1+\hat{h})(kDt-t_{sml})]$$
$$\times cos[w_{IF}kDt + \hat{w}_{D0}kDt + \frac{\hat{a}_D}{2}(kDt)^2]$$
$$\text{for } l = 0,...,(l_a-1) \quad (4a)$$

$$DI_{0L(m,l)}[\hat{t}_s, \hat{w}_{D0}] =$$
$$\sum_{k=mn_{coh}+(L-1)l+l_a}^{mn_{coh}+(L-1)(l+1)+l_a-1} y_k c_{0L}[(1+\hat{h})(kDt-t_{sml})]$$

$$\times cos[\boldsymbol{w}_{IF}k\boldsymbol{D}t + \hat{\boldsymbol{w}}_{D0}k\boldsymbol{D}t + \frac{\hat{\boldsymbol{a}}_D}{2}(k\boldsymbol{D}t)^2]$$
$$\text{for } \ell = \ell_a,...,(\ell_b-1) \tag{4b}$$

$$\boldsymbol{DQ}_{0L(m,\ell)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] =$$
$$\sum_{k=mn_{coh}+L\ell}^{mn_{coh}+L(\ell+1)-1} y_k c_{0L}[(1+\hbar)(k\boldsymbol{D}t - \boldsymbol{t}_{sm\ell})]$$
$$\times sin[\boldsymbol{w}_{IF}k\boldsymbol{D}t + \hat{\boldsymbol{w}}_{D0}k\boldsymbol{D}t + \frac{\hat{\boldsymbol{a}}_D}{2}(k\boldsymbol{D}t)^2]$$
$$\text{for } \ell = 0,...,(\ell_a-1) \tag{4c}$$

$$\boldsymbol{DQ}_{0L(m,\ell)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] =$$
$$\sum_{k=mn_{coh}+(L-1)\ell+\ell_a}^{mn_{coh}+(L-1)(\ell+1)+\ell_a-1} y_k c_{0L}[(1+\hbar)(k\boldsymbol{D}t - \boldsymbol{t}_{sm\ell})]$$
$$\times sin[\boldsymbol{w}_{IF}k\boldsymbol{D}t + \hat{\boldsymbol{w}}_{D0}k\boldsymbol{D}t + \frac{\hat{\boldsymbol{a}}_D}{2}(k\boldsymbol{D}t)^2]$$
$$\text{for } \ell = \ell_a,...,(\ell_b-1) \tag{4d}$$

where ℓ_b is the number of partial accumulations. $L = ceil(n_{coh}/\ell_b)$ is the number of samples in each of the first ℓ_a partial integrations, where $ceil(z)$ is the nearest integer to z rounding towards $+\infty$ and where $\ell_a = \ell_b + n_{coh} - L\ell_b$. The number of samples in the last $\ell_b - \ell_a = L\ell_b - n_{coh}$ partial integrations is L-1. Given these integrals, the coherent correlation accumulations of eqs. (2a) and (2b) can be approximated as

$$I_{0L(m)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] \cong \sum_{\ell=0}^{\ell_b-1} \boldsymbol{DI}_{0L(m,\ell)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] \tag{5a}$$

$$Q_{0L(m)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] \cong \sum_{\ell=0}^{\ell_b-1} \boldsymbol{DQ}_{0L(m,\ell)}[\hat{t}_s, \hat{\boldsymbol{w}}_{D0}] \tag{5b}$$

if one uses the following definition for the approximate code start time:

$$\boldsymbol{t}_{sm\ell} = \left(\frac{1+\hbar_0}{1+\hbar}\right)\hat{t}_s + \left(\frac{\hbar-\hbar_0}{1+\hbar}\right)[mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t$$
$$-\frac{\hat{\boldsymbol{x}}}{2(1+\hbar)}\{[mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t - \hat{t}_s\}^2$$
$$\text{for } \ell = 0,...,(\ell_a-1) \tag{6a}$$

$$\boldsymbol{t}_{sm\ell} =$$
$$\left(\frac{1+\hbar_0}{1+\hbar}\right)\hat{t}_s + \left(\frac{\hbar-\hbar_0}{1+\hbar}\right)[mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t$$
$$-\frac{\hat{\boldsymbol{x}}}{2(1+\hbar)}\{[mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t - \hat{t}_s\}^2$$
$$\text{for } \ell = \ell_a,...,(\ell_b-1) \tag{6b}$$

This approximate code start time corrects, on average, for the differences between the candidate code Doppler shift in eqs. (2a) and (2b) and that used in eqs. (4a)-(4d). These differences arise because the latter equations use

the constant fractional Doppler shift $\hbar$ in place of the time-varying fractional shift $\hbar_0 + \hat{\boldsymbol{x}}(k\boldsymbol{D}t - \hat{t}_s)$. This is the only approximation in going from eqs. (2a) and (2b) to eqs. (5a) and (5b). It is needed in order to allow the evaluation of the accumulations in eqs. (4a)-(4d) using FFT techniques. Given the very small values typical of $\hbar_0$, $\hat{\boldsymbol{x}}$, and $\hbar$, the principal effect of this approximation is to reduce the correlation power slightly, and this loss can be kept to within reasonable limits by bounding the number of samples L in each accumulation.

The partial accumulations in eqs. (4a) and (4b) can be calculated using circular correlation techniques with zero-padding, as explained in Refs. 5-7. Suppose that one defines

$$\boldsymbol{Dz}_{0L(m,\ell)}[n, \hat{\boldsymbol{w}}_D] = \boldsymbol{DI}_{0L(m,\ell)}[\hat{t}_{sm\ell n}, \hat{\boldsymbol{w}}_{D0}]$$
$$+ j\boldsymbol{DQ}_{0L(m,\ell)}[\hat{t}_{sm\ell n}, \hat{\boldsymbol{w}}_{D0}] \tag{7}$$

where $j = \sqrt{-1}$ and where the code start time $\hat{t}_{sm\ell n}$, which is a function of the index n, is defined by

$$\hat{t}_{sm\ell n} = \left(\frac{1+\hbar}{1+\hbar_0}\right)n\boldsymbol{D}t + \left(\frac{\hbar_0-\hbar}{1+\hbar_0}\right)[mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t$$
$$+\frac{\hat{\boldsymbol{x}}}{2(1+\hbar_0)}\{[mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t - \hat{t}_{sm\ell n}\}^2$$
$$\text{for } \ell = 0,...,(\ell_a-1) \tag{8a}$$

$$\hat{t}_{sm\ell n} =$$
$$\left(\frac{1+\hbar}{1+\hbar_0}\right)n\boldsymbol{D}t + \left(\frac{\hbar_0-\hbar}{1+\hbar_0}\right)[mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t$$
$$+\frac{\hat{\boldsymbol{x}}}{2(1+\hbar_0)}\{[mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t - \hat{t}_{sm\ell n}\}^2$$
$$\text{for } \ell = \ell_a,...,(\ell_b-1) \tag{8b}$$

These formulas are implicit because $\hat{t}_{sm\ell n}$ also appears in the last term on the right-hand side of each equation. Reasonable approximations are to use $\hat{t}_{sm\ell n} = n\boldsymbol{D}t$ on the right-hand sides. Improved approximations can be generated by iterating these formulas a second time with $\hat{t}_{sm\ell n}$ from the first iterations used in the right-hand sides during the second iterations. This method of approximating $\hat{t}_{sm\ell n}$ is very accurate when $\hat{\boldsymbol{x}}$ is very small, which is usually the case.

FFT and IFFT operations can be used to calculate $\boldsymbol{Dz}_{0L(m,\ell)}[n, \hat{\boldsymbol{w}}_D]$ at multiple n values simultaneously. These calculations take the form:

$$\begin{bmatrix} C_0 \\ C_1 \\ \vdots \\ C_{N-1} \end{bmatrix} = conj\{FFT(\begin{bmatrix} c_{0L(mn_{coh}+L[\ell+1]-n_0-N)} \\ c_{0L(mn_{coh}+L[\ell+1]-n_0-N+1)} \\ \vdots \\ c_{0L(mn_{coh}+L[\ell+1]-n_0-1)} \end{bmatrix})\}$$

$$\text{for } \ell = 0,...,(\ell_a{-}1) \qquad (9a)$$

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = FFT\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ y_{bb(mn_{coh}+L\ell)} \\ y_{bb(mn_{coh}+L\ell+1)} \\ \vdots \\ y_{bb(mn_{coh}+L[\ell+1]-1)} \end{bmatrix} \right)$$

$$\text{for } \ell = 0,...,(\ell_a{-}1) \qquad (9b)$$

$$\begin{bmatrix} \boldsymbol{Dz}_{0L(m,\ell)}[n_0,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{Dz}_{0L(m,\ell)}[n_0{+}1,\hat{\boldsymbol{w}}_{D0}] \\ \vdots \\ \boldsymbol{Dz}_{0L(m,\ell)}[n_0{+}N{-}L,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{Dz}_{discard(1)} \\ \vdots \\ \boldsymbol{Dz}_{discard(L{-}1)} \end{bmatrix} = IFFT\left(\begin{bmatrix} C_0\,Y_0 \\ C_1\,Y_1 \\ C_2\,Y_2 \\ \vdots \\ C_{N-1}\,Y_{N-1} \end{bmatrix} \right)$$

$$\text{for } \ell = 0,...,(\ell_a{-}1) \qquad (9c)$$

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = FFT\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ y_{bb(mn_{coh}+[L-1]\ell+\ell_a)} \\ y_{bb(mn_{coh}+[L-1]\ell+\ell_a+1)} \\ \vdots \\ y_{bb(mn_{coh}+[L-1][\ell+1]+\ell_a-1)} \end{bmatrix} \right)$$

$$\text{for } \ell = \ell_a,...,(\ell_b{-}1) \qquad (9d)$$

$$\begin{bmatrix} \boldsymbol{Dz}_{0L(m,\ell)}[(n_0{+}\ell_a{-}\ell{-}1),\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{Dz}_{0L(m,\ell)}[(n_0{+}\ell_a{-}\ell),\hat{\boldsymbol{w}}_{D0}] \\ \vdots \\ \boldsymbol{Dz}_{0L(m,\ell)}[(n_0{+}\ell_a{-}\ell{+}N{-}L),\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{Dz}_{discard(1)} \\ \vdots \\ \boldsymbol{Dz}_{discard(L{-}2)} \end{bmatrix} =$$

$$IFFT\left(\begin{bmatrix} C_0\,Y_0 \\ C_1\,Y_1 \\ C_2\,Y_2 \\ \vdots \\ C_{N-1}\,Y_{N-1} \end{bmatrix} \right) \text{ for } \ell = \ell_a,...,(\ell_b{-}1) \qquad (9e)$$

where N is the number of FFT points, $y_{bb(k)} = y_k\, exp\{\, j[\boldsymbol{w}_{IF}k\boldsymbol{D}t + \hat{\boldsymbol{w}}_{D0}k\boldsymbol{D}t + 0.5\hat{\boldsymbol{a}}_D(k\boldsymbol{D}t)^2]\}\}$ is the baseband-mixed version of y_k, $c_{0L(i)} = c_{0L}[(1{+}\hat{h})i\boldsymbol{D}t]$, and $conj(z)$ is the complex conjugate of z.

The calculations in eqs. (9a)-(9e) use zero-padding because the PRN code and the baseband data do not obey the periodicity conditions $c_{0L(i)} = c_{0L(i+N)}$ and $y_{bb(i)} = y_{bb(i+N)}$. The leading $N{-}L$ zeros on the right-hand side of eq. (9b) and the leading $N{-}L{+}1$ zeros on the right-hand side of eq. (9d) provide the needed padding. They make the first $N{-}L{+}1$ outputs on the left-hand side of eq. (9c) and the first $N{-}L{+}2$ outputs on the left-hand side of eq. (9e) correspond

to actual $\boldsymbol{Dz}$ values defined in eq. (7). The remaining $\boldsymbol{Dz}_{discard(i)}$ values must be discarded because of wrap-around in the circular correlation calculations [7].

Figure 1 illustrates the timing layout of the FFT data batches used in eqs. (9a)-(9c) for several sub-intervals of a coherent integration interval and for all possible code start times. For illustrative purposes, this figure depicts the c_{0L} code as a periodic linear curve with a negative slope (top graph) and the y_{bb} signal as a straight line with a positive slope (bottom graph). The figure also assumes that $L\ell_b = n_{coh}$ exactly so that $\ell_a = \ell_b$, which obviates the need to consider eqs. (9d)-(9e). Six over-lapping sub-sections of the c_{0L} time history are used to span one entire code period, and each successive sub-section overlaps the preceding sub-section by L samples. These c_{0L} batches are depicted below the top graph of c_{0L} vs. t. The coherent integration interval of the baseband-mixed y_{bb} data is broken into $\ell_b = 5$ non-overlapping sub-intervals of length L samples, one for each partial accumulation. The data from each sub-interval is padded with $N{-}L$ leading zeros to create the 5 FFT data batches shown on the figure below the lower graph. Calculation of the coherent accumulations $I_{0L(m,\ell)}[\hat{t}_s,\hat{\boldsymbol{w}}_{D0}]$ and $Q_{0L(m,\ell)}[\hat{t}_s,\hat{\boldsymbol{w}}_{D0}]$ on a grid of code start times that covers an entire code period is accomplished by performing the calculations of eqs. (9a)-(9c) for each of the 30 distinct combinations of the top plot's 6 overlapping code intervals and the bottom plot's 5 adjacent portions of the y_{bb} coherent integration interval.
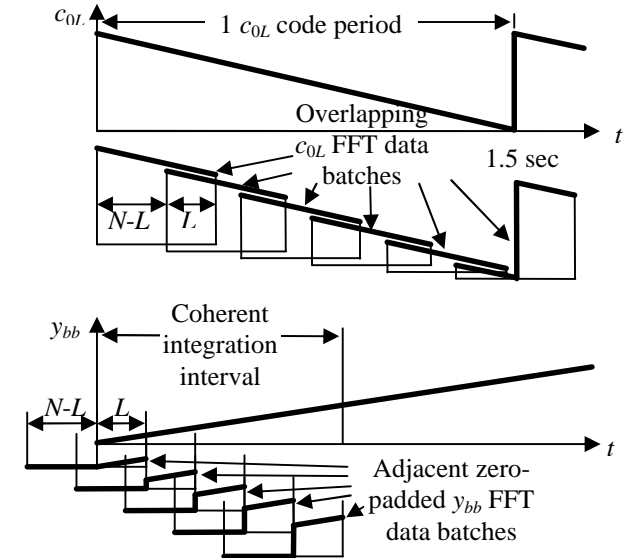

Fig. 1. *Time layout of overlapping FFT data batches of c_{0L} code and zero-padded FFT batches of baseband-mixed signal for use in c_{0L} acquisition calculations.*

One must interpolate the outputs of eqs. (9c) and (9e) onto a common grid of code start times before their real and imaginary parts can be used as $\boldsymbol{DI}_{0L(m,\ell)}[\hat{t}_s,\hat{\boldsymbol{w}}_{D0}]$ and

$DQ_{0L(m,\ell)}[\hat{t}_s,\hat{w}_{D0}]$ in eqs. (5a) and (5b). This is true because the $\hat{t}_{sm\ell n}$ grid values from eqs. (8a) and (8b) that correspond to these outputs vary with m and ℓ due to the difference $\hbar_0-\hbar$ and due to the non-zero value of $\hat{x}$.

The interpolation procedure works as follows. Given $Dz_{0L(m,\ell)}[n_0,\hat{w}_{D0}]$ through $Dz_{0L(m,\ell)}[n_0+N-L,\hat{w}_{D0}]$ from eq. (9c), it first computes $\hat{t}_{sm\ell[n_0]}$ and $\hat{t}_{sm\ell[n_0+N-L]}$, the code start times associated with the first and last outputs. The interpolation makes the simplifying assumption that the intervening values are evenly spaced. This assumption ignores the effect of $\hat{x}$ over the $N-L$ intervals, which is reasonable. The interpolation next computes the fraction $f = (\hat{t}_s - \hat{t}_{sm\ell[n_0]})/(\hat{t}_{sm\ell[n_0+N-L]} - \hat{t}_{sm\ell[n_0]})$. This fraction gets used to compute the indices $n_{lo} = n_0 + floor[f(N-L)]$ and $n_{hi} = n_{lo} +1$, which are the indices of the two code start times with known Dz values that fall on either side of $\hat{t}_s$. Note that the function $floor(z)$ is the nearest integer to z rounded towards $-\infty$. The fraction f also gets used to compute the linear interpolation weights $l_{hi} = mod[f(N-L),1]$ and $l_{lo} = 1 - l_{hi}$. Linear interpolation is then performed between these two times in order estimate the partial accumulations for the code start time $\hat{t}_s$:

$$DI_{0L(m,\ell)}[\hat{t}_s,\hat{w}_{D0}] =$$
$$real\{l_{lo}Dz_{0L(m,\ell)}[n_{lo},\hat{w}_{D0}]+l_{hi}Dz_{0L(m,\ell)}[n_{hi},\hat{w}_{D0}]\}$$
(10a)

$$DQ_{0L(m,\ell)}[\hat{t}_s,\hat{w}_{D0}] =$$
$$imag\{l_{lo}Dz_{0L(m,\ell)}[n_{lo},\hat{w}_{D0}]+l_{hi}Dz_{0L(m,\ell)}[n_{hi},\hat{w}_{D0}]\}$$
(10b)

This interpolation presumes that $0 \leq f \leq 1$, which is equivalent to requiring that $\hat{t}_{sm\ell[n_0]} \leq \hat{t}_s \leq \hat{t}_{sm\ell[n_0+N-L]}$. A similar procedure is used to interpolate the outputs of eq. (9e).

Another situation that requires interpolation arises if the acquisition calculations extend across multiple code periods. In this case, the $\hat{t}_s$ value used to develop the interpolation in eqs. (10a) and (10b) will be a multiple of the code period plus the $\hat{t}_s$ grid value at which the accumulations $I_{0L(m)}[\hat{t}_s,\hat{w}_{D0}]$ and $Q_{0L(m)}[\hat{t}_s,\hat{w}_{D0}]$ and the statistic $P_{0L}[\hat{t}_s,\hat{w}_{D0}]$ are calculated. This latter $\hat{t}_s$ value is no greater than one code period. Interpolation is required because the code period normally is not an integer multiple of the sample period Dt.

In the case of nonzero $\hat{x}$, successive code periods have slightly different lengths. The increment from the original grid value of $\hat{t}_s$ to the interpolation value of $\hat{t}_s$, which is one or more code periods later, must be calculated using an iteration that corrects for the slight changes in code period as time progresses.

Equations (9a), (9c), and (9e) have to be applied several times using a number of different values for n_0 in order to get a range of outputs that span a whole code period's worth of start times $\hat{t}_s$. One might be tempted to increment n_0 by the amount $N-L+1$ between each successive application of these equations in order to generate partial accumulations on a non-overlapping grid of $\hat{t}_{sm\ell n}$ values that have a nominal spacing of Dt. If one does this, then eqs. (10a) and (10b) will have to do extrapolation rather than interpolation in order to determine partial accumulations whenever a $\hat{t}_s$ grid point lies between $\hat{t}_{sm\ell[n_0+N-L]}$ and $\hat{t}_{sm\ell[n_0+N-L+1]}$. This is so because the former time will be associated with the final output of eq. (9c) for one value of n_0, and the latter time will be associated with the initial output of eq. (9c) for the next value of n_0. Interpolation between these two points is not advisable because their partial accumulations are generated in different FFT batches. In order to guarantee that interpolation within a single FFT batch is (almost) always used, the strategy adopted here is to increment n_0 only by $N-L$ between successive applications of eqs. (9c) or (9e).

A very small gap can develop between the $\hat{t}_{sm\ell[n_0+N-L]}$ value associated with the terminal partial accumulation from one evaluation of eq. (9c) and the $\hat{t}_{sm\ell[n_0+N-L]}$ value associated with the initial partial accumulation from the next evaluation. Such a gap can result from a difference in the code Doppler shift for the two intervals. If a $\hat{t}_s$ grid point falls in this gap, then eqs. (10a) and (10b) get used either with $f > 1$ to extrapolate the partial accumulations past the last point of the first FFT batch or with $f < 0$ to extrapolate before the first point of the second FFT batch. Such an extrapolation is permissible because it is very small.

Several nested execution loops are needed in order to carry out the c_{0L} code start time acquisition calculations efficiently for a full range of $\hat{t}_s$ values. The calculations start with an initialization loop, and they finish with a nested set of 3 loops. The appendix presents pseudo code that documents the operations and interactions of these loops.

B. CM Code Start Time Acquisition Calculations

The correlation accumulations for the c_{M0} code acquisition also can be calculated using FFT-based techniques. The main difference of the approach for the c_{M0} code is that the coherent accumulations are fixed with respect to the code rather than the data. The following calculations are used to compute partial coherent accumulations for the c_{M0} code:

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = FFT\left(\begin{bmatrix} y_{bb(n_0+mn_{coh}+L\ell)} \\ y_{bb(n_0+mn_{coh}+L\ell+1)} \\ \vdots \\ y_{bb(n_0+mn_{coh}+L\ell+N-1)} \end{bmatrix}\right)$$
(11a)

$$\begin{bmatrix} C_0 \\ C_1 \\ \vdots \\ C_{N-1} \end{bmatrix} = conj\{FFT(\begin{bmatrix} c_{M0(mn_{coh}+L\ell)} \\ c_{M0(mn_{coh}+L\ell+1)} \\ \vdots \\ c_{M0(mn_{coh}+L[\ell+1]-1)} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix})\}$$

$$\text{for } \ell = 0,...,(\ell_a\text{-}1) \qquad (11b)$$

$$\begin{bmatrix} \boldsymbol{D}z_{M0(m,\ell)}[n_0,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{D}z_{M0(m,\ell)}[n_0+1,\hat{\boldsymbol{w}}_{D0}] \\ \vdots \\ \boldsymbol{D}z_{M0(m,\ell)}[n_0+N-L,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{D}z_{discard(1)} \\ \vdots \\ \boldsymbol{D}z_{discard(L-1)} \end{bmatrix} = IFFT(\begin{bmatrix} C_0 Y_0 \\ C_1 Y_1 \\ C_2 Y_2 \\ \vdots \\ C_{N-1} Y_{N-1} \end{bmatrix})$$

$$\text{for } \ell = 0,...,(\ell_a\text{-}1) \qquad (11c)$$

$$\begin{bmatrix} C_0 \\ C_1 \\ \vdots \\ C_{N-1} \end{bmatrix} = conj\{FFT(\begin{bmatrix} c_{M0(mn_{coh}+[L-1]\ell+\ell_a)} \\ c_{M0(mn_{coh}+[L-1]\ell+\ell_a+1)} \\ \vdots \\ c_{M0(mn_{coh}+[L-1][\ell+1]+\ell_a-1)} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix})\}$$

$$\text{for } \ell = \ell_a,...,(\ell_b\text{-}1) \qquad (11d)$$

$$\begin{bmatrix} \boldsymbol{D}z_{M0(m,\ell)}[n_0+\ell-\ell_a,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{D}z_{M0(m,\ell)}[n_0+\ell-\ell_a+1,\hat{\boldsymbol{w}}_{D0}] \\ \vdots \\ \boldsymbol{D}z_{M0(m,\ell)}[n_0+\ell-\ell_a+N-L+1,\hat{\boldsymbol{w}}_{D0}] \\ \boldsymbol{D}z_{discard(1)} \\ \vdots \\ \boldsymbol{D}z_{discard(L-2)} \end{bmatrix} =$$

$$IFFT(\begin{bmatrix} C_0 Y_0 \\ C_1 Y_1 \\ C_2 Y_2 \\ \vdots \\ C_{N-1} Y_{N-1} \end{bmatrix}) \quad \text{for } \ell = \ell_a,...,(\ell_b\text{-}1) \qquad (11e)$$

where $c_{M0(i)} = c_{M0}[(1+\hat{\boldsymbol{h}})i\boldsymbol{D}t]$. The first N-L+1 entries of the output vector on the right-hand side of eq. (11c) correspond to valid partial accumulations, but the last L-1 entries must be discarded due to the circular correlation's wrap-around effect. The outputs of eq. (11e) have one additional valid partial accumulation and one less output that must be discarded because the number of non-zero code samples in eq. (11d) is only L-1 as compared to L in eq. (11b).

Each valid partial accumulation $\boldsymbol{D}z_{M0(m,\ell)}[n,\hat{\boldsymbol{w}}_{D0}]$ on the left-hand side of eq. (11c) or eq. (11e) corresponds to the approximate CM code start time:

$$\hat{t}_{sm\ell n} \cong n\boldsymbol{D}t + \left(\frac{\hbar_0-\hbar}{1+\hbar_0}\right)[mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t$$
$$+ \frac{\hat{\boldsymbol{x}}}{2(1+\hbar_0)}\{[n+mn_{coh}+L\ell+(L-1)/2]\boldsymbol{D}t - \hat{t}_{sm\ell n}\}^2$$
$$\text{for } \ell = 0,...,(\ell_a\text{-}1) \qquad (12a)$$

$$\hat{t}_{sm\ell n} \cong n\boldsymbol{D}t + \left(\frac{\hbar_0-\hbar}{1+\hbar_0}\right)[mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t$$
$$+ \frac{\hat{\boldsymbol{x}}}{2(1+\hbar_0)}\{[n+mn_{coh}+(L-1)\ell+\ell_a+(L-2)/2]\boldsymbol{D}t - \hat{t}_{sm\ell n}\}^2$$
$$\ell = \ell_a,...,(\ell_b\text{-}1) \qquad (12b)$$

These formulas are similar to the CL code start time formulas in eqs. (8a) and (8b). They are implicit because $\hat{t}_{sm\ell n}$ appears on both the left- and right-hand sides. Reasonable approximations are to input $\hat{t}_{sm\ell n} = n\boldsymbol{D}t$ on the right-hand sides and to use the equations to compute improved approximations for $\hat{t}_{sm\ell n}$. Further improvements can be generated via second iterations that use the first iterations' $\hat{t}_{sm\ell n}$ outputs as inputs on the right-hand sides. The calculations in eqs. (11a)-(11e) approximate the CM code fractional Doppler shift as being constant with the value $\hbar$. The code start time calculations in eqs. (12a) and (12b) correct for the average difference between this code Doppler shift and the time-varying code Doppler shift assumed in the CM equivalent of eqs. (2a) and (2b). The other principal effect of this approximation is a slight loss in correlation power, which can be kept small by enforcing an upper bound on L, which is equivalent to enforcing a lower bound on ℓ_b.

Figure 2 illustrates the relationships between the c_{M0} code, the y_{bb} data, and the FFT batches for the c_{M0} code acquisition calculations in much the same way that Fig. 1 illustrates these relationships for the c_{0L} code acquisition. The example makes the simplifying assumption that $L\ell_b = n_{coh}$ so that eqs. (11d) and (11e) need not be considered because $\ell_a = \ell_b$. The coherent integration interval is defined with respect to the c_{M0} code (top graph), and that interval is broken up into $\ell_b = 8$ partial intervals. The L c_{M0} samples from each of these intervals get padded with N-L trailing zeros in order to create a data batch of length N. The baseband signal gets broken up into overlapping intervals. Each interval includes N samples of y_{bb} data, and it overlaps the preceding interval by L samples (bottom graph).

The CM coherent accumulation calculations proceed from this point much as the CL calculations. The code start times corresponding to the first and last partial accumulations on the left-hand side of eq. (11c) or eq. (11e) are calculated using two iterations of eq. (12a) or eq. (12b), these times are used to interpolate the accumulations to any given start time grid point, $\hat{t}_s$, as in

eqs. (10a) and (10b), and the resulting interpolated partial accumulations are summed to produce complete coherent integrals, as in eqs. (5a) and (5b).
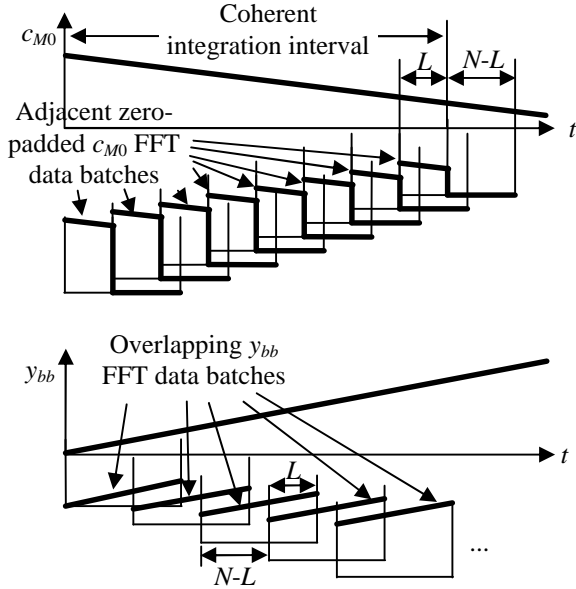

Fig. 2. Time layout of zero-padded FFT data batches for c_{M0} code and overlapping FFT batches of baseband-mixed signal for use in c_{M0} acquisition calculations.

The CM acquisition restricts itself to use coherent integration intervals that do not cross the c_{M0} code's start/stop time. This restriction avoids the possibility of a correlation power loss due to a sign change in a transmitted navigation data symbol $d[t]$. This restriction translates into the following conditions on M and n_{coh}: Either $Mn_{coh} < \{0.020/[(1+\hbar)Dt]\}+1$ or $n_{coh} = ceil([\{0.020/[(1+\hbar)Dt]\}+1]/M_{indep}) - 1$. The former condition applies when the entire acquisition interval is no greater than the length of a single navigation data symbol, $0.020/(1+\hbar)$ sec. The latter condition applies when the acquisition spans more than one data symbol. In this case, there are $M_{indep} < M$ coherent integration intervals per navigation data symbol.

The c_{M0} code start time acquisition uses several loops to compute a statistic like that of eq. (3) for a grid of candidate code start times. A pair of nested loops and a second independent loop are used to initialize code and signal FFTs, and the main calculations are performed by a set of 3 nested loops. Pseudo code for these loops is presented in the appendix.

C. Comparison Between CL and CM Code Start Time Acquisition Calculations

The structure of the CL code start time acquisition has two advantages over that of the CM acquisition. First, a shorter data span can be used for a given set of coherent and non-coherent integration lengths because there is no

need to shift the integration intervals in order to maintain a fixed relationship to each trial value of the code start time. Second, the nested loops which calculate the acquisition statistic do not need to include a complicated calculation in order to determine the range of y_{bb} data batches that cover the entire grid of code start times for a given partial coherent accumulation interval.

D. Power Loss Due to Code Doppler Shift Approximation

The approximate code Doppler shift used in the FFT-based calculations has two principal effects. Recall that the fractional code Doppler shift is approximated by the constant value $\hbar$ while the candidate Doppler shift is $h(t) = \hbar_0 + \hat{x}(t-\hat{t}_s)$. The differences between these Doppler shifts cause timing errors in the code acquisition calculations and power loss. The timing errors get corrected by using the code start time calculations in eqs. (8a) and (8b) or in eqs. (12a) and (12b) along with a code start time interpolation as in eqs. (10a)-(10b).

The power loss can be bounded by limiting the length of each partial coherent integration interval, L. Power loss occurs when the code chips in the y_k data do not align exactly with the chips in the CL or CM code replica. The timing error corrections cause the chips to line up on average. A power loss occurs if there are deviations from this average alignment over the span of a partial coherent accumulation. If the maximum allowable power loss due to code Doppler deviations is DS dB, then the following lower bound on l_b ensures that L will be small enough to keep the power loss from exceeding this value:

$$l_b \geq \frac{1.023\times10^6 n_{coh}Dt \underset{(t-\hat{t}_s)}{max}/\hbar_0 + \hat{x}(t-\hat{t}_s) - \hbar/}{3[1-\sqrt{1-\frac{4}{3}(1-10^{-0.1DS})}]} \quad (13)$$

where this formula is valid for the range $0 < DS \leq 4.77$ dB. Recall that l_b is the number of partial intervals into which each coherent integration interval gets broken up. As an example, consider a typical case in which $DS = 1$ dB and $\underset{(t-\hat{t}_s)}{max} \hbar_0 + \hat{x}(t-\hat{t}_s) - \hbar/ = 3\times10^{-6}$. The bound is $l_b \geq n_{coh}Dt/0.1447$ in this case. Thus, partial coherent integration intervals less than 0.1447 sec will cause correlation power losses due to the error in the FFT's assumed code Doppler shift of less than 1 dB.

IV. FFT FREQUENCY SHIFTING WHEN SEARCHING AT MULTIPLE FREQUENCIES

A. Using a Doppler Shift Grid Spacing Equal to the FFT Frequency Spacing

If the acquisition calculations of Section III must be repeated for several values of the initial carrier Doppler shift $\hat{w}_{D0}$, then circular frequency shifting can be used to

reduce the number of FFT operations [6]. The frequency-domain representations of y_{bb} in eqs. (9b), (9d), and (11a) give the complex Fourier coefficients at multiples of the frequency $Dw_{fft} = 2p/(NDt)$. This fact can be used to develop a relationship between the FFTs of two y_{bb} signals that are computed using two different initial carrier Doppler shifts. One is $\hat{w}_{D0}$, and the other is $\hat{w}_{D0} + iDw_{fft}$, where i is an integer number of frequency increments. The relationship is:

$$
\begin{bmatrix}
\bar{Y}_{mod(-i,N)} \\
\bar{Y}_{mod(-i+1,N)} \\
\vdots \\
\bar{Y}_{N-1} \\
\bar{Y}_0 \\
\bar{Y}_1 \\
\vdots \\
\bar{Y}_{mod(N-i-1,N)}
\end{bmatrix}
exp\{jnDtiDw_{fft}\} =
$$

$$
FFT\left(
\begin{bmatrix}
\bar{y}_{bb(n)} \, exp\{jnDtiDw_{fft}\} \\
\bar{y}_{bb(n+1)} \, exp\{j(n+1)DtiDw_{fft}\} \\
\vdots \\
\bar{y}_{bb(n+N-1)} \, exp\{j(n+N-1)DtiDw_{fft}\}
\end{bmatrix}
\right) \quad (14)
$$

where $mod(x,y) = x - y \times floor(x/y)$. The baseband signal $\bar{y}_{bb(k)}$ in eq. (14) is the y_k signal mixed to baseband using the carrier Doppler shift $\hat{w}_{D0} + \hat{a}_D t$. Thus, the input argument to the FFT operation in this equation is the y_k signal mixed to baseband using the carrier Doppler shift $\hat{w}_{D0} + iDw_{fft} + \hat{a}_D t$. The FFT coefficients $\bar{Y}_0$ through $\bar{Y}_{N-1}$ on the left-hand side of this equation are the outputs of the FFT that uses $\bar{y}_{bb(n)}$ through $\bar{y}_{bb(n+N-1)}$ as the input to the FFT.

The main point of eq. (14) is that one can re-use the outputs of eqs. (9b), (9d), or (11a) to get the FFT of a baseband mixed version of y_k that uses a different initial carrier Doppler shift than was used in the original evaluation of the equation. Re-use is accomplished by using a circular index shift of the $\bar{Y}_k$ outputs, as defined on the right-hand side of eq. (14), along with multiplication of the results by a single complex exponential. This takes $6N$ floating point operations, which is normally much less than the $5N(1+\log_2 N)$ operations that are required to compute the baseband signal at the new Doppler shift and perform the FFT from scratch. This approach can be used to significantly reduce the number of executions of eqs. (9b), (9d), or (11a). Note that the zero padding in eqs. (9b) and (9d) does not affect the fundamental applicability of eq. (14) to the re-use of these two equations.

B. Using a Doppler Shift Grid Spacing Equal to a Fraction of the FFT Frequency Spacing

The frequency spacing Dw_{fft} between different baseband mixing frequencies may be too large for acquisition calculations. This spacing allows for a maximum frequency error of $dw = 0.5Dw_{fft} = p/(NDt)$. An acquisition frequency error leads to a correlation power loss over a coherent integration interval. For a frequency error dw and a coherent integration interval of T, the signal power gets scaled down by the loss factor $[\sin(dwT/2)/(dwT/2)]^2$. A partial coherent integration interval of L samples corresponds to $T_{partial} = LDt$, and the corresponding maximum loss factor is $\{\sin[pL/(2N)]/[pL/(2N)]\}^2$. If $L = N/2$, then the maximum possible signal power loss during a partial coherent accumulation is 0.91 dB.

If the maximum possible power loss is unacceptably large, then additional frequency points must be added to the Doppler shift grid. A good idea is to add points with the spacing Dw_{fft}/p, where p is an integer divisor.

Circular frequency shifting can be used to reduce the number of FFT operations on this modified grid by using the following strategy: The algorithm does actual baseband mixing and FFT operations for the p initial carrier Doppler shifts $\hat{w}_{D0}$, $\hat{w}_{D0} + Dw_{fft}/p$, $\hat{w}_{D0} + 2Dw_{fft}/p$, ..., $\hat{w}_{D0} + (p-1)Dw_{fft}/p$. Circular frequency shifting as in eq. (14) is then used to generate the FFT corresponding to the initial Doppler shift $\hat{w}_{D0} + [i+(k/p)]Dw_{fft}$ from the FFT corresponding to $\hat{w}_{D0} + kDw_{fft}/p$. This can be done for any integer i and for any integer k in the range 0 to $(p-1)$.

V. USE OF COARSE AND FINE FREQUENCY GRIDS

Additional savings in the number of calculations can be had when searching over multiple frequencies if the coherent integration intervals are split up into several partial intervals, as in eqs. (5a) and (5b) when ℓ_b is larger than 1. A savings in the operations count can be achieved by calculating the partial accumulations of eqs. (4a)-(4d) only on a coarse grid of initial Doppler carrier frequencies. These grid points are $\hat{w}_{D0} + [i+(k/p)]Dw_{fft}$ for $k = 0:(p-1)$ and for i ranging far enough to cover the entire frequency search space. The worst-case power loss factor on this grid for a full coherent accumulation is $\{\sin[Dw_{fft}n_{coh}Dt/(4p)]/[Dw_{fft}n_{coh}Dt/(4p)]\}^2$, and it may represent too much loss. Consider an example in which $p = 2$ coarse frequency grid points per FFT frequency interval, $\ell_b = 5$ partial accumulations per coherent interval, and $L = ceil(n_{coh}/\ell_b) = N/2$ samples per partial accumulation interval. The power loss of a partial accumulation due to the maximum possible frequency error on this grid is only 0.22 dB, but the maximum power loss of a full accumulation is 6.54 dB. The only way to avoid this power loss is to use a finer frequency grid spacing for the full accumulations.

Reference 6 presents a technique whereby partial coherent accumulations at one frequency can be used to approximate partial coherent accumulations at another

frequency for use in the full accumulation calculations of eqs. (5a) and (5b). The approximation takes the form of a rotation that corrects for the average phase error between the two frequencies during the partial accumulation interval:

$$DI_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0b}] \cong$$
$$DI_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0a}] \, cos[(\hat{w}_{D0b} - \hat{w}_{D0a})\bar{t}_{(m,\ell)}]$$
$$- DQ_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0a}] \, sin[(\hat{w}_{D0b} - \hat{w}_{D0a})\bar{t}_{(m,\ell)}]$$

$$(15a)$$

$$DQ_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0b}] \cong$$
$$DI_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0a}] \, sin[(\hat{w}_{D0b} - \hat{w}_{D0a})\bar{t}_{(m,\ell)}]$$
$$+ DQ_{0L(m,\ell)}[\hat{t}_s, \hat{w}_{D0a}] \, cos[(\hat{w}_{D0b} - \hat{w}_{D0a})\bar{t}_{(m,\ell)}]$$

$$(15b)$$

where $\hat{w}_{D0a}$ is the initial Doppler shift at which the partial accumulations are known, $\hat{w}_{D0b}$ is the new Doppler shift at which they are to be calculated, and $\bar{t}_{(m,\ell)}$ is the mid-point of the partial accumulation interval. Note that $\bar{t}_{(m,\ell)} = Dt[mn_{coh}+L\ell + (L-1)/2]$ for $\ell = 0,...,(\ell_a-1)$ and $\bar{t}_{(m,\ell)} = Dt[mn_{coh}+(L-1)\ell + \ell_a + (L-2)/2]$ for $\ell = \ell_a,...,(\ell_b-1)$.

This strategy saves the $5N\log_2 N$ operations of an IFFT calculation at the expense of the $6(N-L+1)$ operations that execute eqs. (15a) and (15b) for $N-L+1$ different code start times. Normally $L \cong N/2$, which implies that eqs. (15a) and (15b) reduce the number of operations from $5N\log_2 N$ to about $3N$. For $N = 2^{18}$ FFT points this reduces the required operations by a factor of 30, and for $N = 2^{12}$ the savings factor is 20.

Similar formulas apply for the CM partial accumulations except that $\bar{t}_{(m,\ell)}$ gets modified by adding $\hat{t}_s$ to it in order to account for the fact that the accumulation intervals are tied to the code rather than to the y_k data. Note, however, that it is allowable to add a constant bias to $\bar{t}_{(m,\ell)}$ if the bias is independent of ℓ. Such a bias only causes a rotation of the coherent accumulation vector $\{I_{M0(m)}[\hat{t}_s, \hat{w}_{D0b}]; Q_{M0(m)}[\hat{t}_s, \hat{w}_{D0b}]\}$. A rotation has no effect on the square of this vector's length, which is what gets summed over all of the coherent accumulation intervals to form the acquisition statistic. Therefore, it is allowable to omit $\hat{t}_s$ from the $\bar{t}_{(m,\ell)}$ computation during the CM acquisition calculations.

The acquisition search needs to calculate search statistics on a grid of frequencies. Given a desired frequency search range, the grid should be chosen to span this range and to have a fine enough grid spacing to bound the maximum frequency error so that the worst-case power loss during a coherent integration is not too large. If the grid spacing is Dw_{fine}, then the worst-case power loss factor is $[sin(Dw_{fine}n_{coh}Dt/4)/(Dw_{fine}n_{coh}Dt/4)]^2$. The worst-case power loss is 3.92 dB if $Dw_{fine} = 2p/(n_{coh}Dt)$, 0.91 dB

if $Dw_{fine} = p/(n_{coh}Dt)$, 0.40 dB if $Dw_{fine} = 2p/(3n_{coh}Dt)$, and so on. Thus, an upper bound on the power loss translates into an upper bound on Dw_{fine}.

The acquisition algorithm also uses a coarse frequency grid with spacing Dw_{fft}/p, where the integer p is chosen to limit the worst-case power loss during a partial accumulation interval, as discussed in Section IV. This grid also spans the desired frequency search range. If $Dw_{fft}/p \leq Dw_{fine}$, then the coarse grid can double as the fine grid, and there is no need to use the rotation technique of eqs. (15a) and (15b). If $Dw_{fft}/p > Dw_{fine}$, on the other hand, then the search works with both the coarse grid and the fine grid. It uses the FFT frequency shifting techniques of Section IV to compute the partial coherent accumulations on the coarse grid. It uses the rotation calculations of eqs. (15a) and (15b) to determine the partial accumulations for each fine frequency grid point from those of the nearest coarse grid point. It uses these latter partial accumulations in its acquisition calculations on the fine frequency grid.

Figure 3 shows an example of a coarse frequency grid and its corresponding fine frequency grid. The coarse grid is shown at the top of the figure. Five of the frequencies marked by solid vertical lines on the top grid are frequencies at which the FFT of y_{bb} can be computed using circular-frequency shifting, as in eq. (14). The sixth frequency marked by a solid vertical line, the one near the center that is labeled as a y_{bb} FFT point, is the only one in this set at which an actual FFT of y_{bb} gets calculated. Note how the spacing between these frequencies equals Dw_{fft}. The six coarse-grid frequencies marked by the dotted lines on the top grid share a similar relationship. There are only two such sets of coarse frequency grid points because $p = 2$ for this example. The fine frequency grid in the lower half of the figure has a much closer spacing, $Dw_{fine} \cong Dw_{fft}/5$, because the number of partial coherent accumulations, ℓ_b, is greater than 1. Consider the coarse frequency grid point marked $\hat{w}_{D0a}$ and indicated by a solid dark-grey line on the top grid. The partial accumulations at this frequency can be rotated using eqs. (15a) and (15b) to approximate the partial accumulations at the 5 different $\hat{w}_{D0b}$ points indicated by dark-grey lines on the fine frequency grid in the lower half of the figure.

The frequency-domain techniques illustrated by Fig. 3 reduce the total number of FFT and IFFT operations needed to search at multiple carrier Doppler shifts. The figure contains 60 fine frequency grid points and 12 coarse frequency grid points. Brute-force application of block-processing in this example would require 60 FFT and 60 IFFT operations, one FFT/IFFT pair for each fine frequency grid point. The techniques of Sections IV and V reduce the required number of FFTs to 2 and the required number of IFFTs to 12. This represents a

reduction of the operations count by a factor of roughly $(2\times60)/(2+12) = 8.6$.
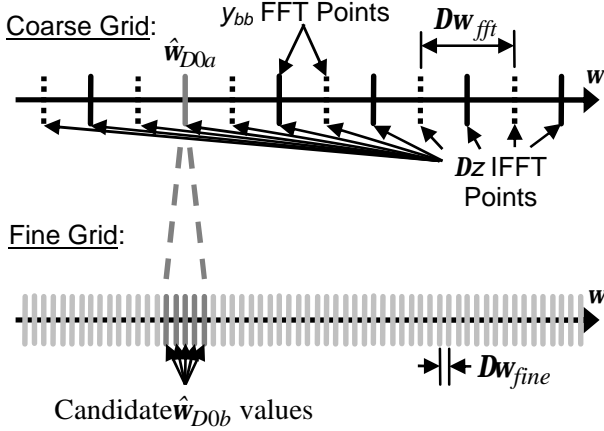

Fig. 3. An example of coarse and fine frequency grids that shows the coarse FFT and IFFT frequencies and the pairing of a coarse-grid frequency with a set of 5 fine-grid frequencies.

VI. OPERATIONS COUNTS AND MEMORY REQUIREMENTS

The CL and CM search algorithms can be expensive in terms of the required number of operations and the required amount of memory. The dominant terms in these two costs have been calculated as functions of acquisition algorithm parameters. This section presents formulas for these terms and analyzes them.

A. Operations Counts

The sum of the dominant terms in the count of the floating point operations needed to implement the CL acquisition algorithm is

$$
\begin{aligned}
n_{calcL} &= n_{codeffi}(5N\log_2 N + 6N) + pM\ell_b(5N\log_2 N + 7L) \\
&\quad + n_{coarse}(3N + M\{\ell_b[6N + n_{codeffi}(5N\log_2 N+6N)]\}) \\
&\quad + n_{tsL}\,n_{fine}M(4+24\ell_b) \\
&\cong n_{coarse}M\ell_b n_{codeffi}(5N\log_2 N + 6N) \\
&\quad + n_{tsL}\,n_{fine}M(24\ell_b)
\end{aligned}
\tag{16}
$$

where $n_{codefft} = 1.5/[\boldsymbol{D}t(N\text{-}L)]$ is the number of FFT's of batches of the c_{0L} PRN code that the algorithm calculates, n_{coarse} is the number of carrier Doppler shifts on the coarse grid, $n_{tsL} = 1.5345\times10^6/\boldsymbol{D}c$ is the number of code start times $\hat{t}_s$ that make up the acquisition search grid, $\boldsymbol{D}c$ is the $\hat{t}_s$ grid spacing in chips, and n_{fine} is the number of carrier Doppler shifts on the fine grid. The number n_{calcL} counts each real scalar multiplication, division, addition, subtraction, and trigonometric evaluation as one floating point operation. It assumes that the acquisition considers code start times which span the entire 1.5 sec period of the c_{0L} code.

The sum of the dominant terms for the number of floating point operations in the CM acquisition algorithm is

$$
\begin{aligned}
n_{calcM} &= M_{indep}\ell_b(5N\log_2 N + N + 6L) \\
&\quad + pn_{ybat}(5N\log_2 N + 8N) \\
&\quad + n_{coarse}[3N + M\ell_b \boldsymbol{D}n_{ybatavg}(5N\log_2 N + 12N)] \\
&\quad + n_{tsM}\,n_{fine}M(4+23\ell_b) \\
&\cong pn_{ybat}(5N\log_2 N) \\
&\quad + n_{coarse}M\ell_b \boldsymbol{D}n_{ybatavg}(5N\log_2 N + 12N) \\
&\quad + n_{tsM}\,n_{fine}M(23\ell_b)
\end{aligned}
\tag{17}
$$

where $n_{tsM} = 20.46\times10^3/\boldsymbol{D}c$ is the number of code start times on the acquisition search grid and where

$$
M_{indep} = min(M, \frac{0.020}{n_{coh}\boldsymbol{D}t})
\tag{18a}
$$

$$
n_{ybat} = ceil\left(\frac{Mn_{coh} + \dfrac{0.020}{\boldsymbol{D}t} - N}{N-L}\right) + 1
\tag{18b}
$$

$$
\boldsymbol{D}n_{ybatavg} = \frac{0.020}{\boldsymbol{D}t(N-L)} + 1
\tag{18c}
$$

M_{indep} is the number of coherent accumulation intervals that span unique sections of the c_{M0} code. n_{ybat} is the number of over-lapping batches of y_k data that must get FFT'd in order to do the entire acquisition calculation. $\boldsymbol{D}n_{ybatavg}$ is the average number of over-lapping batches of y_k data whose FFT's must be used with a single partial coherent integration interval in order to span the full set of possible code start times.

B. Optimization of Operations Counts

The operations counts in eqs. (16) and (17) can be optimized by varying N and ℓ_b subject to a maximum practical bound on N and the minimum bound on ℓ_b from eq. (13). The value of L is determined from ℓ_b according to the formula $L = ceil(n_{coh}/\ell_b)$. This optimization assumes a fixed number of coherent integration intervals M, a fixed number of fine Doppler shift grid points n_{fine}, and a fixed number of code start times, n_{tsL} or n_{tsM}. The number of coarse grid Doppler shift points, n_{coarse}, and the number of frequencies for y_k baseband mixing and FFT operations, p, both get optimized as functions of N and ℓ_b: $p = ceil[(n_{coh}p)/(2N\ell_b\boldsymbol{g})]$ and $n_{coarse} = max\{1, ceil[p_{mod}\boldsymbol{w}_{range}N\boldsymbol{D}t/(2p)]\}$. The quantity $\boldsymbol{g}$ is chosen to yield a desired value of $[(sin\boldsymbol{g})/\boldsymbol{g}]^2$, which represents a lower limit on the power loss factor due to a worst-case frequency error. The quantity $\boldsymbol{w}_{range}$ is the range of initial carrier Doppler shifts over which the acquisition must search. The quantity

$$
p_{mod} = \begin{cases} p & \text{if } p > 1 \\ 1/floor\left(\dfrac{2N\ell_b\boldsymbol{g}}{n_{coh}\boldsymbol{p}}\right) & \text{if } p = 1 \end{cases}
\tag{19}
$$

is a modified version of the frequency count p. The actual

coarse grid spacing is $Dw_{fft}/p_{mod} = 2p/(ND tp_{mod})$. If p_{mod} is less than 1, then $1/p_{mod}$ is an integer, and the coarse grid spacing is an integer multiple of Dw_{fft}.

A heuristic search method has been developed for optimizing the algorithm operation counts n_{calcL} and n_{calcM}. The search starts by trying an N value that is equal to its maximum practical power of 2, and it works downwards in an outer search loop, decreasing N by a factor of 2 on each successive iteration. This outer loop stops when a decrease in N produces an increase in the operations count. An inner loop searches to find the optimal l_b value for each trial value of N. This inner search steps through l_b values starting from the lower limit and incrementing by 1. It stops at the optimal value l_{bopt} when it finds that $n_{calc}(N, l_{bopt}) \leq n_{calc}(N, l_{bopt}+Dl_b)$ for $Dl_b = 1, 2, 3, 4,$ and 5. This termination criterion checks that 5 values of l_b past l_{bopt} do not produce lower operations counts. It is needed because $n_{calcL}(N, l_b)$ and $n_{calcM}(N, l_b)$ are not always smooth functions of l_b.

A typical case is depicted in Fig. 4, which plots n_{calcL} vs. l_b for several different values of N. Parameters corresponding to this case are $Dt = 0.2$ μsec, $n_{tsL} = 6.138 \times 10^6$ code start time grid points, $n_{fine} = 21$ Doppler shift grid points spread out over a frequency range of $w_{range}/(2p) = 42$ Hz, and $M = 4$ coherent summation intervals each of length $n_{coh} = 1.25 \times 10^6$ (i.e., $n_{coh}Dt = 0.25$ sec). The figure shows that $n_{calcL}(N, l_b)$ is not always a smooth function of l_b. The kinks in the curves are caused by integer transitions in the outputs of the rounding functions $ceil()$ and $floor()$. The figure also shows that the optimum n_{calcL} value decreases as N increases. Although not shown in the figure, the optimum n_{calcL} continues to decrease until N reaches 2^{23}, at which point it is only 60% of the optimal value at $N = 2^{19}$. Even in this situation, however, the required number of floating point operations is very large, $n_{calcL} = 1.3 \times 10^{11}$.

C. Memory Requirements

The memory requirements of the two algorithms are as follows:

$$n_{storeL} = 2N(n_{codefft} + Ml_b) + 2n_{tsL} \, ceil(n_{fine}/n_{coarse}) + n_{tsL} \, n_{fine} \quad (20a)$$

$$n_{storeM} = 2N(M_{indep}l_b + n_{ybat}) + 2n_{tsM} \, ceil(n_{fine}/n_{coarse}) + n_{tsM} \, n_{fine} \quad (20b)$$

where n_{storeL} is for the CL algorithm and n_{storeM} is for the CM algorithm. These are the leading terms in the amount of storage required measured as a number of real scalars. These numbers exclude the storage required by called functions, most notably, the FFT and IFFT functions. The last term in each of these expressions is the amount of memory needed to store the output detection statistic on the desired grid of code start times and initial carrier Doppler shifts. The remaining terms are temporary

storage that is needed during execution.

The interim storage requirements can be rather large, especially for the CL acquisition if it uses a long total integration time Ml_bLDt. In some situations, it may be advisable to store the temporary and final results on disk rather than in RAM. It is possible to structure the algorithms so that disk storage can be used without undue thrashing of data in and out of RAM.
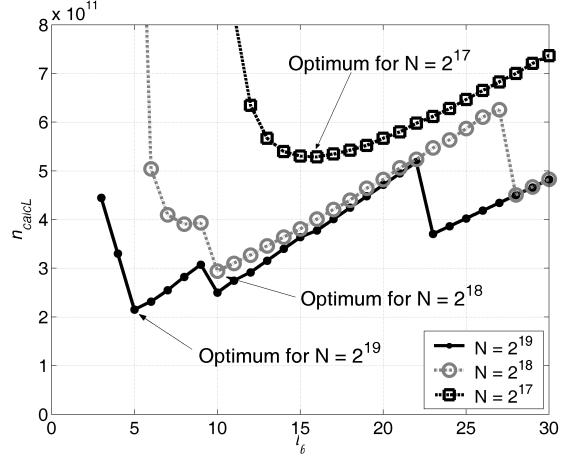

Fig. 4. *Operations count for a CL acquisition calculation, n_{calcL}, as a function of the number of partial intervals of each coherent integration interval, l_b, for several different FFT batch lengths, N.*

VII. COMPUTATIONAL RESULTS

A. Generation and Recording of Simulated L2 Civilian Signals

The algorithms described in Sections III-V have been encoded in MATLAB, and they have been tested using simulated GPS L2 civilian signals. It has been necessary to use simulated data because no Block IIR-M satellites, with the new L2C signals, have been launched at the time of this research. Two types of simulation data have been generated. One type has come from an off-line receiver simulation that runs in MATLAB and that simulates the output of a dual-frequency L1/L2 RF front end. The other data has been recorded using a custom L2 RF front end that has been connected to the output of a SPIRENT Model STR 4760 GPS simulator running SimGEN for Windows version V2.41.

The off-line MATLAB simulator generates the L1 C/A code and the L2 CM/CL code, and it models many realistic signal effects. These include range acceleration variations, carrier amplitude variations, receiver clock drift, receiver thermal noise, 2-bit digitization, interference from other GPS signals, differential code delay and carrier advance due to ionospheric effects, and navigation data symbols. The only significant effect that is not simulated is the PRN code distortion due to the

finite widths of the RF front end's bandpass filters at L1 and L2. In addition to RF front end data, the simulator outputs "truth" values for carrier phase, code phase, carrier amplitude, etc.

The data from the SPIRENT hardware-in-the-loop simulator is mixed to a lower frequency, digitized, and recorded using a prototype L2 civilian RF front end that is connected to a computer data acquisition system. Several stages of mixing and filtering are involved, with some of the stages being implemented digitally in the computer subsequent to sampling. The final output of the RF front end is a real signal whose L2 carrier frequency has been mixed to 1.25 MHz. It is sampled at 5 MHz with a 16-bit digitization. The 3 dB bandwidth of the front end's narrowest filter is 2.2 MHz, and the front end's noise figure is slightly less than 1 dB. The front end has been connected to the simulator output by a cable whose length is less than 0.6 m.

B. Simulation Results for Geosynchronous Acquisition Cases

The SPIRENT simulator was set up to transmit signals that will be broadcast by the Block IIR-M satellites, namely the C/A and P(Y) codes on L1 and the L2C and P(Y) codes on L2. The simulation scenario was based on the actual GPS constellation circa October 2001, and it modeled a user in a Geostationary orbit. The simulated link model included GPS signal path losses and a GPS satellite antenna gain pattern based on a Block IIA L1 antenna (same antenna model used for both L1 and L2 signals). The user antenna was modeled as omni-directional (zero dB gain).

The simulation included a single "strong" L2C signal from the main lobe of PRN 1 with a peak received power of approximately –169 dBW (received carrier-to-noise ratio of C/N_0 = 36 dB-Hz[*]) and 7 side-lobe signals with received power levels ranging from –185 to –181 dBW (received C/N_0 values ranging from 20 dB-Hz to 24 dB-Hz). Additional side-lobe signals were present at -186 dBW (C/N_0 values of about 19 dB-Hz); however, these additional signals had unrealistically high power levels due to the –20 dB attenuation limit of the SPIRENT simulator.

Three different signal acquisitions have been carried out using this data. The three signals that have been acquired are PRN 1's main-lobe signal, a side-lobe signal with C/N_0 = 21 dB-Hz from PRN 2, and one of the unrealistic side-lobe signals with C/N_0 = 19 dB-Hz from PRN 25. All three signals have been acquired using the CM code, and

PRN 25 has also been acquired using the CL code.

As an example, consider the PRN 2 acquisition using the CM code. The acquisition used M = 200 coherent integration intervals, each with length $n_{coh}Dt$ = 0.020 sec, i.e., each with a length of one full navigation data symbol. Thus, the entire acquisition was based on 4 seconds worth of data. The algorithm has been run with $N = 2^{18}$ points per FFT data batch, and with f_b = 1 partial interval per coherent integration interval. The frequency search range was W_{range} = 0 because the initial carrier frequency was known to within better than 50 Hz = $1/(n_{coh}Dt)$. The resolution of the code start time search was Dc = 0.25 chips. The number of floating point operations executed by the search was n_{calcM} = 1.2×10^{10}. The acquisition required storage for n_{storeM} = 65.8×10^6 real numbers, with most of that being used for temporary storage. The formulas at the end of Section II indicate that the brute-force calculation would require 1.6×10^{13} floating point operations. Thus, the new algorithm is faster by a factor of 1.3×10^3.

The success of this acquisition is indicated by the plot in Fig. 5. It shows the P_{M0} acquisition statistic as a function of the code start time error $\hat{t}_s - t_s$. Also shown is the acquisition threshold that yields a 0.01% probability of false alarm. P_{M0} at the true code start time exceeds this threshold by a large amount. This is as expected because the probability of a missed detection is less than 10^{-11} for this case. Although the "truth" code phase is not known, the correctness of this acquisition has been confirmed by doing two acquisitions separated by 60 seconds. The code start time from the later acquisition agrees with that of the earlier acquisition to within 0.18 chips after taking into account the code Doppler shift that is predicted by the carrier Doppler shift. This agreement confirms the success of both acquisitions.
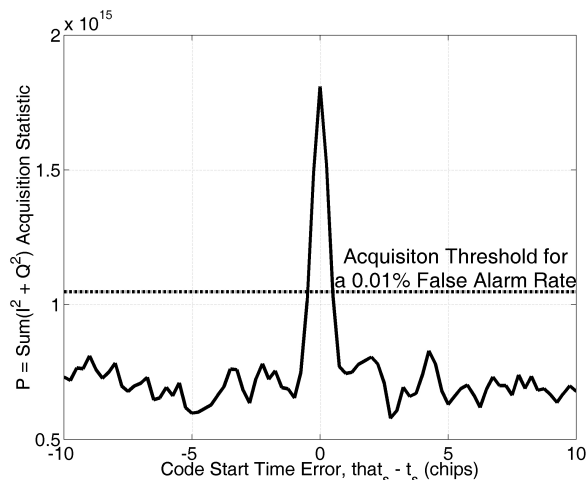

Fig. 5. The P_{M0} acquisition statistic vs. the code start time error for CM acquisition of a side-lobe signal at geosynchronous altitude, PRN 2 with C/N_0 = 21 dB-Hz.

C. Simulation Result for an Acquisition at an Altitude of 17 Earth Radii

The CL acquisition algorithm has been tested on an extreme case that attempts to detect side-lobe signals for a user at an altitude of 17 Earth radii. This level of sensitivity would enable autonomous absolute and relative navigation of spacecraft in very highly elliptical orbits, even extending out to the cis-lunar libration point (approximately 50 Earth radii) [8]. Data has been generated using the off-line MATLAB simulation with the signal power level set to yield $C/N_0 = 9$ dB-Hz, representative of the side-lobe signals that would be present for a user at 17 Earth radii. If the receiving system uses a patch antenna with a gain of 4.5 dB and an RF front end with a noise figure of 1 dB, then this C/N_0 level allows continuous visibility of Block IIR-M GPS signals with line-of-sight (LOS) anywhere between the edge of Earth and 30 deg away from the center of the transmission antenna's field of view (FOV). Beyond 30 degrees, the gain of the Block IIR-M satellite antenna array varies significantly depending on the transmission azimuth angle; however, some signals are actually present above 9 dB-Hz with line-of-sight vectors as far as 85 deg from the FOV center.

Note that CL acquisiton is required for this case. The limited coherent integration interval of the CM code, 0.020 sec, implies that over 140 sec of data would have to be processed in order to achieve a probability of missed detection lower than 10% while maintaining a false alarm probability below 0.1 %. The lack of data bits on the CL code allows a much longer coherent integration interval, which strengthens the acquisition power. The example case has used a coherent integration interval of $n_{coh}Dt =$ 2.5 sec and integration over $M = 8$ such intervals for a total acquisition data span of 20 sec. This acquisition has achieved a probability of missed detection equal to 1.8×10^{-5} with a false alarm probability of only 0.01 %. Thus, CL acquisition is much more practical for very weak civilian L2 signals.

Other parameters of this acquisition example are as follows: The PRN number is 3, the sample period is $Dt =$ 0.0820 µsec, the number of data points per FFT batch is $N = 2^{20}$, each coherent integration interval is split up into $\ell_b =$ 59 partial intervals, the frequency search range is $W_{range}/(2p) = 0.83$ Hz, and the code start time resolution is $Dc = 0.25$ chips. The execution requires $n_{calcL} = 2.2 \times 10^{12}$ floating point operations and storage of $n_{storeL} = 1.2 \times 10^9$ real numbers, most of which is temporary storage. If brute-force acquisition were carried out on this problem, then the computational cost would be 7.5×10^{16} floating point operations. The new algorithm accelerates the execution speed by a factor of 33×10^3.

Figure 6 demonstrates the success of this acquisition. The peak acquisition statistic is well above the 0.01% false alarm threshold. Its code start time is correct to within 0.094 chips. This is within the resolution of the calculation, $Dc/2 = 0.125$ chips. The carrier Doppler shift has been correctly determined: notice how the two curves with w_{D0} errors of $\pm$ 0.2 Hz have peaks that are much smaller than the peak at the true carrier Doppler shift. The fineness of the carrier Doppler resolution is the result of using a long coherent integration interval, $n_{coh}Dt = 2.5$ sec, which gives a frequency resolution that is better than $1/(2.5 \text{ sec}) = 0.4$ Hz.

This acquisition calculation has made use of carrier Doppler rate aiding. Recall that this aiding takes the form of a known value of a_D. The simulated "truth" value for this quantity has been used in the acquisition, $a_D = -6.43$ rad/sec^2. Use of this truth value presumes the ability to model the orbit of the user receiver and the use of a stable oscillator in the receiver. The simulation used receiver clock stability parameters that are typical of an ovenized crystal oscillator.
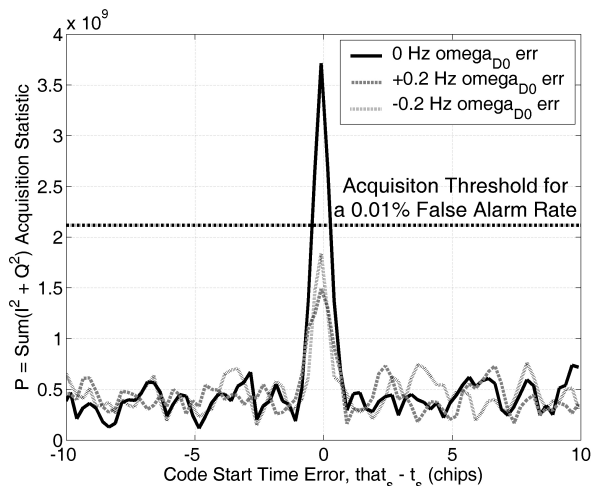

Fig. 6. The P_{0L} acquisition statistic vs. the code start time error at 3 different carrier Doppler shift errors for CL acquisition of a side-lobe signal from PRN 3 at an altitude of 17 Earth radii, $C/N_0 = 9$ dB-Hz.

VIII. SUMMARY AND CONCLUSIONS

This paper has developed efficient block processing acquisition algorithms for the new GPS CL and CM signals that will be broadcast on the L2 frequency once the first Block IIR-M satellites are launched. Two distinct algorithms have been designed, one for the CL code and one for the CM code because the latter carries navigation data while the former does not. These algorithms use FFT techniques to calculate correlation accumulations simultaneously for multiple PRN code start times. The algorithms use zero-padding in an overlap-and-discard approach to adjust for the fact that the PRN code period does not equal the FFT batch length because of limitations on this length. The algorithms use interpolation to adjust for differences between the code start time grid points of

individual FFT batches and those of the overall calculation. If the acquisition search encompasses a range of carrier Doppler shifts, then the algorithms use circular frequency shifting and rotations from a coarse frequency grid to a fine frequency grid in order to reduce the required number of FFT and IFFT operations. The algorithms include an aiding option whereby a known rate of change of the carrier Doppler shift gets used in the baseband mixing process.

These algorithms have been shown to work well on simulations of weak civilian L2 signals that might be acquired by a user receiver aboard a high Earth orbiting satellite that lies outside of the main lobe of the GPS transmission antenna. A signal from a hardware-in-the-loop simulator with a carrier-to-noise ratio of 21 dB-Hz has been acquired using the CM algorithm operating on 4 seconds worth of data, and a 9 dB-Hz signal from an off-line simulator has been acquired using the CL algorithm and 20 seconds worth of data. The CM algorithm executed more than 1,300 times faster than an equivalent brute-force algorithm, and the CL algorithm executed 33,000 times faster.

APPENDIX: PSEUDO CODE FOR ACQUISITION CALCULATIONS

A. Pseudo code for acquisition of the c_{0L} code start time for a single assumed carrier Doppler shift:

for (all overlapping c_{0L} code FFT batches spaced N-L samples apart)

    perform eq. (9a) FFT and conjugation operations on batch and store result

end

$P_{0Lvec} = 0$

for $m = 0$:(M-1); i.e., for all M coherent integration intervals

    $I_{0L(m)vec} = 0$

    $Q_{0L(m)vec} = 0$

    for $\ell = 0$:(ℓ_b-1) i.e., for all ℓ_b partial coherent integration intervals

        mix non-overlapping span of y_k data to baseband, pad with zeros, perform eq. (9b) or (9d) FFT, and store result

        for (all overlapping c_{0L} code FFT batches spaced N-L samples apart)

            mix FFT of y_{bb} partial coherent interval with FFT of c_{0L} code batch and IFFT result as in eqs. (9c) or (9e)

            Interpolate resulting partial accumulations onto partial grid of code start times

as in eqs. (10a) and (10b) to determine some elements of $DI_{0L(m,\ell)vec}$ and $DQ_{0L(m,\ell)vec}$ and add results to corresponding elements of $I_{0L(m)vec}$ and $Q_{0L(m)vec}$ as in eqs. (5a) and (5b).

        end

    end

    $P_{0Lvec} = P_{0Lvec} + [I_{0L(m)vec}]^2 + [Q_{0L(m)vec}]^2$

end

The subscript ()$_{vec}$ in the foregoing pseudo code has been appended to various quantities in order to indicate that a vector of quantities is involved in which each element corresponds to a different $\hat{t}_s$ grid point for which the acquisition calculations are being performed.

B. Pseudo code for acquisition of the c_{M0} code start time for a single assumed carrier Doppler shift:

for (all coherent integration intervals within a single c_{M0} code period)

    for $\ell = 0$:(ℓ_b-1) i.e., for all ℓ_b partial coherent integration intervals

        perform eq. (11b) or eq. (11d) FFT and conjugation operations on non-overlapping partial coherent span of c_{M0} code with trailing padded zeros and store result

    end

end

for (all overlapping y_{bb} batches needed in entire calculation)

    mix y_k batch to baseband, perform eq. (11a) FFT of y_{bb}, and store result

end

$P_{M0vec} = 0$

for $m = 0$:(M-1); i.e., for all M coherent integration intervals

    $I_{M0(m)vec} = 0$

    $Q_{M0(m)vec} = 0$

    for $\ell = 0$:(ℓ_b-1) i.e., for all ℓ_b partial coherent integration intervals

        retrieve relevant conjugated FFT of zero-padded c_{M0} code span

        for (all overlapping y_{bb} batches needed for this partial coherent interval)

            mix FFT of y_{bb} data batch with FFT of

> partial coherent interval of zero-padded c_{M0} code and IFFT result as in eq. (11c) or eq. (11e)
>
> > Interpolate resulting partial accumulations onto partial grid of code start times in a manner similar to eqs. (10a) and (10b) to determine some elements of $\boldsymbol{DI}_{M0(m,\ell)vec}$ and $\boldsymbol{DQ}_{M0(m,\ell)vec}$ and add results to corresponding elements of $I_{M0(m)vec}$ and $Q_{M0(m)vec}$ similar to eqs. (5a) and (5b).
>
> > end
>
> > end
>
> > $P_{M0vec} = P_{M0vec} + [I_{M0(m)vec}]^2 + [Q_{M0(m)vec}]^2$
>
> end

The range of the required y_k/y_{bb} data batches in the second initialization loop is calculated as a function of three inputs: the range of code start times being considered, the number of coherent integration intervals being summed to compute the detection statistic, and the length of each coherent integration interval. Similarly, calculation of the correct range for the inner-most loop in the last set of nested loops involves a determination of the range of y_{bb} FFT batches that are needed in order to span the full range of the acquisition's c_{M0} code start times for a given partial accumulation interval. The specific formulas for these ranges are omitted for the sake of brevity.

C. Modifications to pseudo code to implement circular frequency shifting

The circular frequency-shifting approach of Section IV.A can be implemented efficiently by modifying the execution loop structures presented above. In the CL calculation's loop structure, the final set of 3 nested loops gets placed within a 4th outer loop, which executes for all initial carrier Doppler shifts, $\hat{w}_{D0} + i\boldsymbol{Dw}_{fft}$ for the desired range of i. In addition, the first operation inside the $\ell=0:(\ell_b-1)$ loop gets modified to replace the baseband mixing, zero-padding, and FFT calculations of eqs. (9b) and (9d) with the circular frequency shifting and complex multiplication of eq. (14). An additional modification is that a new pair of nested initialization loops is needed in order to do the y_k baseband mixing, zero-padding, and FFT operations of eqs. (9b) and (9d) for the initial carrier Doppler shift $\hat{w}_{D0}$. These new loops resemble the outer two loops of the CL pseudo code's original 3 nested loops.

Similar modifications apply to the pseudo code for the CM acquisition calculations with circular frequency shifting, but with two differences. First, there are no additional initialization loops for baseband mixing with the carrier Doppler shift $\hat{w}_{D0}$ because the CM pseudo

code already includes such an initialization loop. Second, the inner-most loop in the final set of 4 nested loops is the place where the eq.-(14) calculations are used to synthesize the FFT of the y_{bb} that corresponds to the appropriate initial Doppler shift.

Additional modifications to the looping structures are needed in order to use circular frequency shifting on a grid of initial Doppler shifts of the form $\hat{w}_{D0} + [i+(k/p)]\boldsymbol{Dw}_{fft}$, as described in Section IV.B. The changes to the looping structure for CL acquisition and for CM acquisition are similar. The approach is to wrap an outer loop around all of the calculations except for the code FFT initialization loops. This outer loop executes for $k = 0:(p-1)$, and each iteration works with a different nominal initial carrier Doppler shift $\hat{w}_{D0} + k\boldsymbol{Dw}_{fft}/p$. The other modification concerns the loop that cycles through different initial Doppler shifts by using circular frequency shifting. The actual frequencies through which it cycles are $\hat{w}_{D0} + [i+(k/p)]\boldsymbol{Dw}_{fft}$ for a fixed k value and an appropriate range of i values.

Each algorithm must store and update two-dimensional arrays of acquisition statistics P, coherent accumulations I and Q, and partial coherent accumulations $\boldsymbol{DI}$ and $\boldsymbol{DQ}$. One index of each array keeps track of the various different code start times that the acquisition is considering. The other index keeps track of the initial carrier Doppler shifts that are being considered.

D. Modifications to pseudo code to implement coarse-grid/fine-grid calculations

The search algorithms' looping execution structure must be further modified in order to efficiently use the coarse-grid/fine-grid approach of Section V. The CL and CM acquisition algorithms' looping structures both get a new inner-most loop in the final set of nested loops. This loop cycles through all points on the fine frequency grid whose nearest coarse frequency grid point is the point being considered in the coarse frequency loop that is outside of this loop. This inner loop performs the eqs. (15a) and (15b) rotations of the partial accumulations that are calculated in the next outer loop for the given coarse frequency grid point. It also calculates the code start time interpolations which are appropriate for the code Doppler shift that corresponds to the fine-grid carrier Doppler shift.

REFERENCES

1. Fontana, R.D., Cheung, W., Novak, P.M., and Stansell, T.A., "The New L2 Civil Signal," *Proceedings of the ION GPS 2001*, Sept. 11-14, 2001, Salt Lake City, UT, pp. 617-631.

2. Long, A., Kelbel, D., Lee, T., Garrison, J., and Carpenter, J.R., "Autonomous Navigation Improvements for High-Earth Orbiters Using GPS," Paper no. MS00/13, *Proceedings of the 15th International Symposium on Spaceflight Dynamics*, CNES, June 26-30, 2000, Biarritz, France, pp. unnumbered.

3. Moreau, M.C., Axelrad, P., Garrison, J.L., and Long, A., "GPS Receiver Architecture and Expected Performance for Autonomous Navigation in High Earth Orbits," *Navigation*, Vol. 47, No. 3, Fall 2000, pp. 191-204.

4. Van Dierendonck, A.J., "GPS Receivers," in *Global Positioning System: Theory and Applications, Vol. I*, Parkinson, B.W. and Spilker, J.J. Jr., eds., American Institute of Aeronautics and Astronautics, (Washington, 1996), pp. 329-407.

5. Tsui. J.B.Y., *Fundamentals of Global Positioning System Receivers, A Software Approach*, J. Wiley & Sons, (New York, 2000), pp. 2-3, 133-164.

6. Psiaki, M.L., "Block Acquisition of Weak GPS Signals in a Software Receiver," *Proceedings of the ION GPS 2001*, Sept. 11-14, 2001, Salt Lake City, UT, pp. 2838-2850.

7. Yang, C., "FFT Acquisition of Periodic, Aperiodic, Puncture, and Overlaid Code Sequences in GPS," *Proceedings of the ION GPS 2001*, Sept. 11-14, 2001, Salt Lake City, UT, pp. 137-147.

8. Carpenter, J.R., Folta, D.C., Moreau, M.C., and Quinn, D.A., "Libration Point Navigation Concepts Supporting the Vision for Space Exploration," to appear in *Astrodynamics 2004*, Univelt, 2004.