Block Acquisition of Weak GPS Signals in a Software Receiver

Mark L. Psiaki, *Cornell University*

BIOGRAPHY

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

ABSTRACT

Block algorithms have been developed to acquire very weak Global Positioning System (GPS) coarse/acquisition (C/A) signals in a software receiver. These algorithms are being developed in order to enable the use of weak GPS signals in applications such as geo-stationary orbit determination. The algorithms average signals over multiple GPS data bits after a squaring operation that removes the bits' signs. Methods have been developed to ensure that the pre-squaring summation intervals do not contain data bit transitions. The algorithms make judicious use of Fast Fourier Transform (FFT) and inverse FFT (IFFT) techniques in order to speed up operations. Signals have been successfully acquired from 4 seconds worth of bit-grabbed data with signal-to-noise ratios (SNRs) as low as 21 dB Hz.

INTRODUCTION

Global Positioning System receivers must acquire and track the pseudo-random number (PRN) code and carrier signal from several GPS satellites. A typical receiver channel has two modes of operation: acquisition and tracking. The acquisition mode estimates the PRN code phase and the carrier Doppler shift via a search process. These quantities are then used to initiate tracking, which continuously updates them.

It is difficult to acquire or track a signal when its SNR is very low. Most commercial receivers can use a signal only if its SNR is greater than about 35 to 37 dB Hz [1,2].

An ability to acquire and track weaker signals would enable important applications. GPS-based orbit determination would become more practical for high-altitude spacecraft if signals with SNRs as low as 28 dB Hz could be used [1,3]. Another potential application is in the monitoring of ionospheric scintillations, which can produce very low transient power levels.

The present paper concentrates on the problem of weak signal acquisition in a software receiver. It deals exclusively with the C/A code on the GPS L1 frequency, 1575.42 MHz. A software receiver performs signal acquisition and tracking tasks using a hardware RF front end and software that runs in a general-purpose microprocessor [4]. Software receivers can be used in non-real-time applications, which allows them to apply sophisticated signal processing techniques. Reference 5's orbit determination method includes an example application of a non-real-time software receiver.

GPS signal acquisition is a mature subject. Reference 6 reviews acquisition methods for real-time receivers. Reference 4 presents special block processing algorithms for acquisition in software receivers. Such algorithms significantly enhance the computational efficiency of acquisition by employing FFT techniques to do code correlation. Low-SNR acquisition techniques for real-time receivers are presented in Ref. 7. Reference 8 deals with low-SNR acquisition in a software receiver.

Reference 8 claims that one needs to know the GPS data bits in order to use acquisition intervals that are longer than 0.010 sec, half the span of a data bit. This restriction leads to a lower limit of about 35 dB Hz for acquisition without prior knowledge of the data bits. The paper goes on to show that weaker signals can be acquired if one knows the data bits a priori, and it demonstrates acquisition down to an SNR of about 15 dB Hz for the bit-aided case.

The present work develops acquisition algorithms that do not use a priori knowledge of the data bits and that operate successfully far below the 35 dB Hz SNR threshold. It uses a known technique whereby the

correlation accumulations are averaged, squared, and then averaged further [6]. It augments this technique with either of two schemes that preclude the occurrence of data bit transitions during the pre-squaring summation intervals.

Another contribution is that this work tailors its algorithms to use FFT-based block processing. Approaches are developed that economize on the number of FFTs and IFFTs that need to be computed. Other processing economies are developed as well.

The issue of weak signal acquisition in the presence of strong interfering GPS signals is also addressed. This problem is important because the small cross-correlations between different PRN codes become significant when the acquired signal's power is much lower than that of an interfering signal.

This paper presents acquisition test results that use real data from a GPS bit-grabber that was connected to a roof-top antenna. A bit-grabber is an RF front end that stores its digitized output on a disk for later software processing. Low SNR cases have been created in software by adding simulated colored noise to the experimental bit-grabber data.

The remainder of this paper is divided into 5 main sections plus conclusions. The second section presents a model of the signal, and it reviews a standard block processing FFT-based acquisition algorithm. The third section explains how to extend un-bit-aided acquisition to multiple data bit intervals. The fourth section shows how to use block processing techniques efficiently in un-aided acquisition. The problem of weak signal acquisition in the presence of strong interfering signals is addressed in the fifth section. The sixth section presents semi-experimental results for the various acquisition algorithms and gives guidelines for achieving a successful acquisition.

SIGNAL MODEL AND REVIEW OF BLOCK-PROCESSING SIGNAL ACQUISITION

Intermediate-Frequency RF Signal Model

The software receiver works with the digitized signal that comes out of the RF front end. The following is a model of this signal for the front end that has been used in this study:

$$y_k = A\, d(t_k)\, c[(1+h)(t_k - t_s)]\, cos[w_{IF}t_k - (w_D t_k + f_0)] + n_k \quad (1)$$

In this model y_k is the output of the RF front end at sample time t_k. The constant A is the signal amplitude. The function $d(t)$ is the GPS data stream, which is sequence of +1 and –1 values that switch randomly approximately every 0.020 sec. The function $c(t)$ is the C/A PRN code of the received signal. This is a known pseudo-random sequence of +1 and –1 values that switch at a nominal chipping rate of 1.023 MHz and that repeat every 1023 code chips. Thus, $c(t)$ is periodic with a period of 0.001 sec. The quantity h is the fractional perturbation of the chipping rate due to Doppler shift, and t_s is the start time of the PRN code, which is a measure of its phase. The frequency w_{IF} is the nominal intermediate frequency to which the L1 carrier signal gets mixed in the RF front end, and w_D is the carrier signal's Doppler shift. Neglecting ionospheric effects, the Doppler shift and the chipping rate perturbation are related: $h = w_D/(2p\times1575.42\times10^6)$. The angle f_0 is the initial carrier phase.

The term n_k is the noise. Normally it can be modeled as Gaussian, band-limited colored noise. The coloring comes from the RF front end's band-pass filters. Sometimes it will be helpful to explicitly model the effects on n_k of interfering GPS satellites.

The 50 Hz data bit shifts in $d(t)$ are synchronized with the PRN code $c(t)$. They occur once every 20 PRN code periods, at the times

$$t_m = \frac{0.020m + 0.001i}{1+h} + t_s \qquad \text{for } m = 0,1,2,\ldots \quad (2)$$

where i is an unknown, but fixed integer somewhere in the range from 0 to 19 inclusive.

Review of Signal Acquisition Using Block Processing

The goal of signal acquisition is to estimate t_s and w_D based on the measurements $y_0, y_1, y_2, \ldots, y_k, \ldots$ The standard technique searches for the t_s and w_D values that maximize the correlation between the signal and a reconstruction of it. Suppose that $\hat{t}_s$ and $\hat{w}_D$ are guessed values of the code phase and the Doppler shift and that $\hat{h} = \hat{w}_D/(2p\times1575.42\times10^6)$. Then the necessary correlations are:

$$I = \sum_{k=0}^{N-1} y_k c[(1+\hat{h})(t_k - \hat{t}_s)]\, cos[(w_{IF} - \hat{w}_D)t_k] \quad (3a)$$

$$Q = -\sum_{k=0}^{N-1} y_k c[(1+\hat{h})(t_k - \hat{t}_s)]\, sin[(w_{IF} - \hat{w}_D)t_k] \quad (3b)$$

where N is the maximum N that satisfies $t_{N-1} < t_0 + 0.001/(1+\hat{h})$, which causes $y_0,\ldots,y_{N-1}$ to span a whole PRN code period of approximately 0.001 sec. The quantity I is known as the in-phase accumulation, and Q is called the quadrature accumulation.

The acquisition maximizes $P(\hat{t}_s, \hat{w}_D) = I^2+Q^2$ via brute force search on the 2-dimensional grid of trial points $\hat{t}_s = t_0, t_1, t_2, \ldots, t_{N-1}$ and $\hat{w}_D = w_{Dmin}, w_{Dmin}+Dw_D, w_{Dmin}+2Dw_D,$

…, w_{Dmax}. The limits w_{Dmin} and w_{Dmax} constitute *a priori* knowledge of the possible range of Doppler shifts. If $P(\hat{t}_s, \hat{w}_D)$ exceeds a threshold value that is determined by the noise statistics, then the acquisition algorithm terminates successfully. If all of the P values on the search grid fall below this threshold, then no signal is detected, and the acquisition fails.

Fast Fourier Transform techniques can speed up the correlation calculations in a software receiver. Suppose that $Dt = t_{k+1} - t_k$ = a constant. Then eqs. (3a) and (3b) can be re-written as

$$z(n, \hat{w}_D) = z(t_0 + nDt, \hat{w}_D) = I + jQ = \sum_{k=0}^{N-1} y_k c_{k-n} \exp[-j(w_{IF} - \hat{w}_D)t_k] \qquad (4)$$

where the integer n is an index of the code start time $\hat{t}_{s(n)} = t_0 + nDt$ and where $c_{k-n} = c[(1+\hat{h})(k-n)Dt]$ is a sampled version of the signal's PRN code.

For a given $\hat{w}_D$, one must compute $z(n, \hat{w}_D)$ for all of the code delays in the range $n = 0, \ldots, N-1$. These can be computed simultaneously in a batch (or block) process that makes use of FFT and IFFT calculations. If one uses circular convolution techniques like those of Ref. 4 and if one computes the following FFTs:

$$\begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{bmatrix} = \mathrm{FFT}\left(\begin{bmatrix} y_0 \exp[-j(w_{IF} - \hat{w}_D)t_0] \\ y_1 \exp[-j(w_{IF} - \hat{w}_D)t_1] \\ \vdots \\ y_{N-1} \exp[-j(w_{IF} - \hat{w}_D)t_{N-1}] \end{bmatrix} \right) \text{ and}$$

$$\begin{bmatrix} C_0 \\ C_1 \\ \vdots \\ C_{N-1} \end{bmatrix} = \mathrm{FFT}\left(\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{N-1} \end{bmatrix} \right) \qquad (5)$$

then it is straightforward to show that

$$\begin{bmatrix} z(0, \hat{w}_D) \\ z(1, \hat{w}_D) \\ \vdots \\ z(N-1, \hat{w}_D) \end{bmatrix} = \mathrm{IFFT}\left(\begin{bmatrix} C_0 Y_0 \\ C_{N-1} Y_1 \\ C_{N-2} Y_2 \\ \vdots \\ C_1 Y_{N-1} \end{bmatrix} \right) \qquad (6)$$

This calculation assumes that the sampled PRN code is periodic with period N, i.e., that $c_k = c_{k+N}$ for all k. Code periodicity is enforced by data interpolation, which is defined below.

Block processing techniques speed up the correlation calculations by orders of magnitude. Radix-2 FFT and IFFT algorithms execute in order($N \log_2 N$) operations if N is a power of 2. Non-FFT-based techniques require order(N^2) operations. The savings factor is $N/\log_2(N)$. This factor ranges from 340 to 630 for typical values of N, which range from 2^{12} to 2^{13}.

Data interpolation is used to accomplish two goals: a) to make c_k exactly periodic with period N and b) to make N equal to a power of 2. Suppose that the RF front end samples the C/A code at about 5 MHz. If one interpolates this sampled data using the shorter sample period $Dt_{interp} = 0.001/[8192(1+\hat{h})]$, then there are exactly $N = 8192$ interpolated samples per PRN code period, and the sampled PRN code is exactly periodic: $c_k = c_{k+8192}$ for all k. Linear interpolation of the y_k data has been used in the present study in order to achieve 8192 samples per code period. The effect of this interpolation on the SNR has been checked by comparing it with the SNR for un-interpolated data. No significant SNR degradation has been found.

This paper's remaining analyses assume the use of data interpolation as defined above. They work with redefined sample times $t_k = t_0 + k \times Dt_{interp}$, a redefined $Dt = Dt_{interp}$, and the redefined y_k values that result from interpolation of the RF front end's original outputs onto the new time grid. This implies that the t_k values, Dt, and the y_k values change with changes in $\hat{w}_D$. This is true because $\hat{h}$ changes with $\hat{w}_D$. These variations necessitate re-interpolation of y_k for each value of $\hat{w}_D$. The resulting re-interpolation effort does not add a significant amount of time to the computation if one is careful to use a special-purpose interpolation procedure which takes advantage of the fact that the original sample times are evenly spaced.

Acquisition of Weaker Signals by Block Processing

The acquisition process defined above requires modifications if the SNR is significantly weaker than the nominal terrestrial value of 50 dB Hz. The easiest thing to do is to sum correlations over adjacent PRN code periods. Suppose that

$$z_{\mathbf{l}}(n, \hat{w}_D) = I_{\mathbf{l}} + jQ_{\mathbf{l}} = \sum_{k=N\mathbf{l}}^{N(\mathbf{l}+1)-1} y_k c_{k-n} \exp[-j(w_{IF} - \hat{w}_D)t_k] \qquad (7)$$

is the correlation for the $\mathbf{l}$ th PRN code period. Then the modified acquisition computes

$$z_{sum}(n, \hat{w}_D) = I_{sum} + jQ_{sum} = \sum_{\mathbf{l}=0}^{L-1} z_{\mathbf{l}}(n, \hat{w}_D) \qquad (8)$$

The acquisition procedure searches for a combination of $\hat{t}_{s(n)}$ and $\hat{w}_D$ that yields a value of $P_{sum} = (I_{sum})^2 + (Q_{sum})^2$ which is above a certain threshold. Although $z_{sum}(n, \hat{w}_D)$ for $n = 0, \ldots, N-1$ could be computed by a single

FFT/IFFT calculation over the long time interval from t_0 to $t_0 + L\times 0.001$, it is more efficient to add the results of individual FFT/IFFT-based computations of $z_{\boldsymbol{I}}(n,\hat{w}_D)$, as in eq. (8). Also, one must search on a finer grid of $\hat{w}_D$ values in this modified acquisition because large losses of correlation occur when $\hat{w}_D$ differs from the true Doppler shift by values approaching $1000\times 2\boldsymbol{p}/L$ rad/sec.

The SNR of the $z_{sum}(n,\hat{w}_D)$ calculation can be increased, up to a point, by increasing L, but there is a limit to the useful size of L. This limit is imposed by the presence of unknown data bit transitions, which can entirely null out the signal if they occur in bad places.

Reference 8 deals with the data bit problem by a judicious choice of its acquisition interval. The idea is to limit L to be no greater than 10 and to try two different intervals. The second interval is delayed by 0.010 sec from the first. This guarantees that at least one of these intervals does not contain a data bit transition because transitions are separated by multiples of 0.020 sec. This acquisition strategy is effective for SNR levels as low as 35 dB Hz.

SIGNAL ACQUISITION OVER MULTIPLE DATA BITS WITHOUT A PRIORI BIT INFORMATION

This section describes two methods for acquiring signals with SNRs below 35 dB Hz. These methods use no *a priori* knowledge about the data bits or about their transition times. Both of the algorithms are variants of the technique that is described in Appendix B of Ref. 6. The basic idea is to sum correlations over short intervals, as in eq. (8), to compute the magnitude squared of each such sum, and then to sum these squared magnitudes over longer intervals. This technique allows one to average over intervals that are much longer than half of a data bit.

This technique uses the following test statistic in order to determine whether or not there is a signal present with the code delay $\hat{t}_{s(n)}$ and the Doppler shift $\hat{w}_D$:

$$P_{long}(n,\hat{w}_D) = \sum_{m=0}^{M-1}\left\{\left[\sum_{\boldsymbol{I}=\boldsymbol{I}_m}^{\boldsymbol{I}_m+L-1} I_{\boldsymbol{I}}(n,\hat{w}_D)\right]^2 + \left[\sum_{\boldsymbol{I}=\boldsymbol{I}_m}^{\boldsymbol{I}_m+L-1} Q_{\boldsymbol{I}}(n,\hat{w}_D)\right]^2\right\} \quad (9)$$

There are M short pre-squaring summation intervals. The new test statistic $P_{long}(n,\hat{w}_D)$ is the sum of the individual P_{sum} values from these M intervals. Each short summation interval spans L PRN code periods, and they start at the PRN code periods $\boldsymbol{I}_0, \boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_{M-1}$.

The test statistic in eq. (9) has two advantages. First, one can make M large and yet avoid signal losses that might occur due to summation over data bit transitions. The squaring operations after the inner summations strip the data bits. Second, one need not search for $\hat{w}_D$ over nearly as fine a Doppler shift grid as one would have to use if one summed the entire data batch before squaring. One can use a Doppler shift grid spacing on the order of $1000\times 2\boldsymbol{p}/L$ rad/sec. A much finer spacing, on the order of $1000\times 2\boldsymbol{p}/(L+\boldsymbol{I}_{M-1}-\boldsymbol{I}_0)$ rad/sec, would be needed if the correlations for the entire data interval were summed before squaring.

An important aspect of the technique concerns how to pick the pre-squaring summation intervals, i.e., how to pick L and $\boldsymbol{I}_0, \boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_{M-1}$. The goal is to ensure that no data bit transitions occur during any of these short summation intervals. This goal can be achieved in two ways.

Alternate Half-Bits Method

One interval selection method is an extension of the technique of Ref. 8. It performs two independent acquisition calculations using alternate sets of $\boldsymbol{I}_k$ values; call them Set "a" and Set "b". Set "a" uses $\boldsymbol{I}_0 = 0$, $\boldsymbol{I}_1 = 20$, $\boldsymbol{I}_2 = 40$, …, $\boldsymbol{I}_{M-1} = (M\text{-}1)\times 20$. Set "b" is displaced half a bit later than set "a": $\boldsymbol{I}_0 = 10$, $\boldsymbol{I}_1 = 30$, $\boldsymbol{I}_2 = 50$, …, $\boldsymbol{I}_{M-1} = 10+(M\text{-}1)\times 20$. Each pre-squaring summation interval is half a bit in duration, i.e., $L = 10$. Thus, one of the sets is guaranteed to have no bit transitions in any of its pre-squaring summation intervals.

Figure 1 illustrates this method. It shows a time history of $Q_{\boldsymbol{I}}$ accumulations for a case in which $\hat{t}_s$ and $\hat{w}_D$ have been chosen correctly. All of the signal power is phased in quadrature. The presence of a signal causes the non-zero mean of the absolute value of the $Q_{\boldsymbol{I}}$ time history. GPS data bit transitions cause the 7 jumps of the signal between a positive mean value and a negative mean value. Superimposed on the plot are the Set-"a" and Set-"b" pre-squaring summation intervals. The Set-"a" intervals start at the vertical dashed lines and end at the vertical dotted lines. The Set-"b" intervals are vice versa, and the two sets of intervals are interleaved. All of the data bit transitions occur in pre-squaring intervals from Set "a". The P_{long} statistic of Set "b" would take on the largest value and, therefore, would be used to acquire the signal.

Full-Bits Method with Estimation of Bit Transition Times

This paper's other method of picking the summation intervals involves estimation of the data bit transition times. The technique used here is similar to the optimum bit synchronizer of Ref. 9. The start indices of the pre-squaring summation intervals are set to $\boldsymbol{I}_0 = i$, $\boldsymbol{I}_1 = i+20$, $\boldsymbol{I}_2 = i+40$, …, $\boldsymbol{I}_{M-1} = i+(M\text{-}1)\times 20$, and the duration of each summation is a full bit, $L = 20$. The P_{long} detection statistic is then computed for each i in the set

{0,1,2,…,19}, and the value that gets used for a particular $\hat{t}_s$ and $\hat{w}_D$ is the maximum of P_{long} over all 20 values. If $\hat{t}_s$ and $\hat{w}_D$ have been chosen correctly, then the corresponding optimal i is an estimate of the data bit transition time.

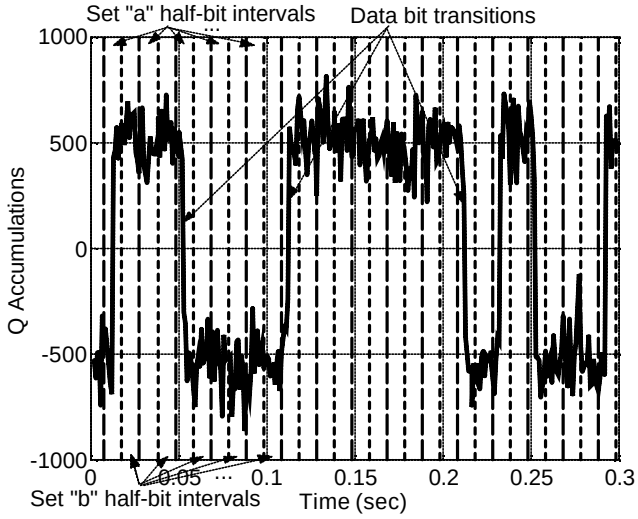

Fig. 1. Example relationship of pre-squaring summation intervals to data bit transitions for the alternate-half-bits method.

Comparison of the Two Algorithms

The principal advantage of the full-bits method over the alternate-half-bits method is that it sums accumulations for twice as much time before squaring. This enables it to detect weaker signals in a fixed batch of data. There is an SNR loss due to squaring. Any SNR gain that can be achieved before squaring is worth double that amount, in dB terms, after squaring. (Strictly speaking, this is true only for false alarm calculations.) The full-bits method achieves 3 dB more gain before squaring because it doubles the pre-squaring summation interval. This translates into a 6 dB gain in the SNR of the acquisition statistic P_{long}.

Another advantage of the full-bits method is its simultaneous estimation of the data bit start times. These are required to decode the navigation message and to infer pseudorange.

The advantage of the half-bits algorithm is its lower complexity. Its code is simpler, it uses less memory, and it executes faster. Even though it performs two separate acquisition calculations, the Set-"a" calculations and the Set-"b" calculations, it still executes faster than the full-bits algorithm when operating on the same amount of raw data.

Hypothesis Testing Analysis of the Acquisition Algorithms

A signal acquisition algorithm can be treated as a hypothesis testing problem, as in Refs. 6 and 10. The signal detection test $P_{long} \geq P_{min}$ is a mixed Bayes/Neyman-Pearson test [10]. In both types of test one effectively compares the data's probability density conditioned on the assumption of no signal with its probability density conditioned on the assumption that there is a signal. The analysis of this particular test assumes that the measurement noise is Gaussian and uncorrelated over different pre-squaring summation intervals, that there are no data bit transitions in the pre-squaring summations, that the occurrence of positive and negative data bits is random and equally likely, and that the initial carrier phase, f_0, is randomly distributed on the interval $[0,2p]$. These assumptions allow one to compute a Bayesian probability distribution for the measured data conditioned on the presence of a signal with a given code phase, Doppler shift, and amplitude. This probability density function is a complicated expression that involves an integral over the random carrier phase. The integrand is an exponential function multiplied by products of hyperbolic cosine functions. The hyperbolic cosine functions result from a Bayesian treatment of the uncertain data bits, as in Ref. 9.

The unknown signal amplitude is removed from the problem by using the technique of generating a locally most powerful Neyman-Pearson hypothesis test (see Ref. 10, pp. 51-55). A locally most powerful test yields the smallest probability of a missed detection for a given false alarm probability in the limit as the signal amplitude approaches zero. Such a test is useful because the weak signal case is the case for which one is most concerned about achieving optimality. The locally optimal test performs a Taylor series expansion of the conditional probability in terms of A, the signal amplitude. This Taylor series is taken about $A = 0$, and the locally optimal test is developed by using the first non-zero coefficient in the series [10]. After some manipulation, the resulting test takes the form $P_{long} > P_{min}$ with P_{long} computed as in eq. (9).

The search over a grid of code delays $\hat{t}_{s(n)}$ and Doppler shifts $\hat{w}_D$ effectively solves an M-ary (as opposed to binary) hypothesis testing problem, as in Ref. 10, p. 61. Each grid point in $[\hat{t}_{s(n)}, \hat{w}_D]$ space constitutes a hypothesis as does the hypothesis that there is no signal. This latter is called the null hypothesis. A reasonable ad hoc approach that is like the M-ary Bayesian approach is to choose as the true hypothesis that combination of $\hat{t}_{s(n)}$ and $\hat{w}_D$ which produces that largest value of P_{long} subject to the constraint that this value exceed P_{min}.

Otherwise, the null hypothesis is chosen, and the system fails to detect a signal. In the full-bits case one also searches over all possible bit start times $i \in \{0,1,2,\ldots 19\}$, but the algorithm is the same otherwise.

One can calculate or bound missed detection probabilities and false alarm probabilities for these two signal detection algorithms. These calculations can be used to design M and P_{min} so as to enforce upper bounds on the probabilities of these two error types. Most of the necessary analyses and equations can be found in Appendix B of Ref. 6. One must keep in mind that M in Ref. 6 corresponds to L in the present paper, that K in Ref. 6 corresponds to M in the present paper, that $(S/N_0)T$ in Ref. 6 corresponds to $(NA)^2/[8(s_{IQ})^2]$ in the present paper, and that λ in Ref. 6 corresponds to $P_{long}/[L(s_{IQ})^2]$ in the present paper. Note that $(s_{IQ})^2$ is the variance of I_λ and of Q_λ that is caused by the measurement noise n_k. Note, also, that eq. (B7) of Ref. 6 has a sign error in the argument of the exponential function: the term $-2Kb$ should be $+2Kb$.

The probability of detection, which equals one minus the probability of missed detection, can be calculated using eq. (B10) in Ref. 6. It depends on the following three quantities: M, the expression $L(NA)^2/[8(s_{IQ})^2]$, and the normalized threshold $P_{min}/[L(s_{IQ})^2]$.

For both of this paper's algorithms, the false alarm probability is a function of only two quantities: M and the normalized threshold $P_{min}/[L(s_{IQ})^2]$. The alternate half-bits method yields a no-signal probability distribution of $P_{long}/[L(s_{IQ})^2]$ that is a chi-squared distribution of degree $2M$, and eq. (B12) of Ref. 6 can be used to calculate the corresponding false alarm probability. In the full-bits case, however, the no-signal probability distribution of $P_{long}/[L(s_{IQ})^2]$ is more complicated. For a given code delay and Doppler shift, there is correlation between the $P_{long}/[L(s_{IQ})^2]$ chi-squared distributions that correspond to different bit start times i. If one maximizes with respect to i, then the distribution of the resulting maximum $P_{long}/[L(s_{IQ})^2]$ cannot be calculated in closed form. Fortunately, one can calculate a Bonferroni-type upper bound on the false alarm probability for this case [11]. It equals 20 times the false alarm probability of eq. (B12) of Ref. 6.

This factor of 20 in the full-bits method causes it to lose some of the benefit of the SNR gain that results from its longer pre-squaring summation interval. This happens because the method's maximization over i tends to amplify the noise when there is no signal present. Fortunately, the SNR loss due to this effect is never more than the 6 dB gain due to the longer pre-squaring summation intervals. Experimental results show that the SNR loss due to maximization with respect to i seems to decrease as the number of post-squaring summations M increases. In a case with M = 199 this loss was only 1 dB.

The allowable false alarm probability must take into account the size of the $[\hat{t}_{s(n)}, \hat{w}_D]$ search space. Suppose that one searches over the full code phase uncertainty of 1023 chips and over a Doppler shift uncertainty range from w_{Dmin} to w_{Dmax}. Suppose, also, that there is no signal present. Then there are approximately $K = 1023\times(w_{Dmax}-w_{Dmin})\times L\times 0.001/(2p)$ statistically independent P_{long} values in this search space. A conservative Bonferroni-type calculation takes the false alarm probability computed from eq. (B12) of Ref. 6 and scales it up by the factor K. One then designs P_{min} so as to enforce a certain bound on this re-scaled probability. For the full-bits method, one scales the eq.-(B12) false alarm probability by the factor $20K$ in order to account for both the K independent P_{long} calculations and the maximization with respect to i. For the alternate half-bits methods, the eq.-(B12) false alarm probability gets scaled up by the factor $2K$. The extra factor of 2 comes from the fact that Sets "a" and "b" together yield 2K statistically independent P_{long} calculations.

Note, one might falsely conclude that $(s_{IQ})^2 = (N/2)(s_n)^2$, where $(s_n)^2$ is the variance of the measurement noise n_k. Normally $(s_{IQ})^2$ is significantly larger than this due to the band-limited nature of n_k. For the bit-grabber used in the present study, the raw data yields $(s_{IQ})^2 = 1.28\ (N/2)(s_n)^2$, and the interpolated data yields $(s_{IQ})^2 = 1.82\ (N/2)(s_n)^2$ when N = 8192 points.

EFFICIENT BLOCK PROCESSING FOR COMPUTATION OF THE ACQUISITION STATISTIC

The acquisition algorithms can require a large amount of computation time. This is especially true if the number of pre-squaring summation intervals, M, is large and if the Doppler shift uncertainty, w_{Dmax}-w_{Dmin}, is large. Block processing can be used to rapidly calculate $[I_\lambda(0,\hat{w}_D),\ldots,I_\lambda(N\text{-}1,\hat{w}_D)]$ and $[Q_\lambda(0,\hat{w}_D),\ldots, Q_\lambda(N\text{-}1,\hat{w}_D)]$ in one batch, but one must still step through $L\times M$ different λ values and many different $\hat{w}_D$ values. Furthermore, if one is using the full-bits method, then one must keep track of 20 running sums of possible P_{long} values at each code delay index n and each Doppler shift grid point $\hat{w}_D$.

Approximation of Accumulations on a Finely-Spaced Doppler Shift Grid

There is a way to economize on the number of FFT and IFFT calculations that must be computed on a large Doppler shift grid. The maximum grid spacing between adjacent Doppler shift trial points is $2p/(0.001L)$ rad/sec, and one may want to use an even finer grid in order to maintain the pre-squaring SNR. Fortunately, one can save FFT calculations by approximating the values of $I_l(n,\hat{w}_D)$ and $Q_l(n,\hat{w}_D)$ on a fine grid based on nearby values from a rough grid.

Suppose that the eq.-(7) accumulations are to be calculated at the Doppler shift $\hat{w}_D = \hat{w}_{Dr} + D\hat{w}_D$. Assume that $\hat{w}_{Dr}$ is a Doppler shift for which accumulations have already been computed and that $D\hat{w}_D$ is a small perturbation. Equation (7) can then be re-written and approximated as follows:

$$
\begin{aligned}
& I_{\boldsymbol{l}}(n,\hat{w}_D) + jQ_{\boldsymbol{l}}(n,\hat{w}_D) \\
&= \sum_{k=N\boldsymbol{l}}^{N(\boldsymbol{l}+1)-1} y_k c_{k-n} \exp[-j(w_{IF} - \hat{w}_{Dr} - D\hat{w}_D)t_k] \\
&= \sum_{k=N\boldsymbol{l}}^{N(\boldsymbol{l}+1)-1} y_k c_{k-n} \exp[-j(w_{IF} - \hat{w}_{Dr})t_k] \bullet \\
&\qquad \exp[jD\hat{w}_D(t_k - \bar{t}_{\boldsymbol{l}})]\exp[jD\hat{w}_D\bar{t}_{\boldsymbol{l}}] \\
&\cong \exp[jD\hat{w}_D\bar{t}_{\boldsymbol{l}}] \sum_{k=N\boldsymbol{l}}^{N(\boldsymbol{l}+1)-1} y_k c_{k-n} \exp[-j(w_{IF} - \hat{w}_{Dr})t_k] \\
&\cong \exp[jD\hat{w}_D\bar{t}_{\boldsymbol{l}}]\{I_{\boldsymbol{l}}(n,\hat{w}_{Dr}) + jQ_{\boldsymbol{l}}(n,\hat{w}_{Dr})\} \qquad (10)
\end{aligned}
$$

where $\bar{t}_{\boldsymbol{l}} = t_{[N(\boldsymbol{l}+0.5)]}$ is the mid point of the $\boldsymbol{l}$ th code-period-length accumulation interval. The approximation made on the third line of eq. (10) is that $\exp[jD\hat{w}_D(t_k - \bar{t}_{\boldsymbol{l}})] \cong 1$. The definition of $\bar{t}_{\boldsymbol{l}}$ guarantees that $|t_k - \bar{t}_{\boldsymbol{l}}| \le 0.0005$ sec. Therefore, the eq.-(10) approximation will be valid if $D\hat{w}_D/(2p) << 2000$ Hz. The approximation is reasonable for $D\hat{w}_D/(2p) \le 500$ Hz and very good for $D\hat{w}_D/(2p) \le 250$ Hz because the average of $\exp[jD\hat{w}_D(t_k - \bar{t}_{\boldsymbol{l}})]$ over the summation interval is very close to 1.

The importance of eq. (10) is that it allows one to restrict the use of expensive FFT and IFFT operations to a rough grid of Doppler shift values $\hat{w}_{Dr0}$, $\hat{w}_{Dr1}$, $\hat{w}_{Dr2}$, … The spacing between these rough grid points can be as large as 500 Hz. The accumulations on a fine Doppler shift grid can then be approximated by using eq. (10) and the I_l and Q_l values of the nearest neighboring point on the rough frequency grid. If the fine grid spacing is 25 Hz and the rough grid spacing is 500 Hz, then there will be a savings factor of 20 in the number of times that FFT and IFFT calculations need to be carried out.

One must account for the differences in code chipping rate between the accumulations on the rough Doppler shift grid and the approximate accumulations on the fine grid. This is especially important for acquisitions that use long data intervals. The non-dimensional difference in the chipping rate is $D\hat{h} = D\hat{w}_D/(2p\times1575.42\times10^6)$. The effect of this difference is to cause the $I_{\boldsymbol{l}}(n,\hat{w}_D)$ and $Q_{\boldsymbol{l}}(n,\hat{w}_D)$ accumulations in eq. (10) to be associated with slightly different code start times:

$$
\{\hat{t}_{s(n)}\}_{new} = \hat{t}_{s(n)} + \frac{D\hat{h}(\bar{t}_{\boldsymbol{l}} - t_0)}{1 + \hat{h}_r + D\hat{h}} \qquad (11)
$$

where $\hat{h}_r = \hat{w}_{Dr}/(2p\times1575.42\times10^6)$ is the fractional chipping rate perturbation associated with the nearest point on the rough Doppler grid. Equation (11) is an approximation that is valid only for $D\hat{h} << 1/1023$, which is true in virtually all foreseeable circumstances.

It is necessary to interpolate accumulations onto a grid of code start times that does not vary with the accumulations' PRN code period $\boldsymbol{l}$. The perturbed code start time in eq. (11) depends on $\boldsymbol{l}$. The grid to choose is $\hat{t}_{s(n)} = t_0 + 0.001n/[N(1 + D\hat{h} + \hat{h}_r)]$ for $n = 0,\ldots, N\text{-}1$. This is accomplished by using eq. (11) to find the integer k such that $\{\hat{t}_{s(n+k)}\}_{new} \le \hat{t}_{s(n)} < \{\hat{t}_{s(n+k+1)}\}_{new}$. One then interpolates linearly between $I_{\boldsymbol{l}}(n+k,\hat{w}_D)$ and $I_{\boldsymbol{l}}(n+k+1,\hat{w}_D)$ in order to find the I accumulation for the estimated code start time $\hat{t}_{s(n)}$. A similar interpolation is used for the Q accumulation. If either $n+k$ or $n+k+1$ is outside of the range 0,1,2,…, N-1, then one uses the periodicity property of the accumulations, $I_{\boldsymbol{l}}(n+N,\hat{w}_D) = I_{\boldsymbol{l}}(n,\hat{w}_D)$ and $Q_{\boldsymbol{l}}(n+N,\hat{w}_D) = Q_{\boldsymbol{l}}(n,\hat{w}_D)$ in order to set up a sensible interpolation.

Worst-Case SNR Loss as a Function of Doppler Shift Grid Spacings

The spacings of the fine and rough Doppler shift grids can affect the SNR of the calculations. The pre-squaring SNR suffers a loss whenever there is a difference between the signal's Doppler shift and the Doppler shift of the grid point. The loss due to rough-grid errors occurs because the summand of eq. (4) drifts around a circle in (I,Q) space during the summation. Fine grid frequency errors cause the $I_{\boldsymbol{l}}$ and $Q_{\boldsymbol{l}}$ accumulations of eq. (9) to drift around a circle during that equation's pre-squaring summation. If the drift goes too far during a summation, then a significant amount of signal power will be lost. The worst-case combined SNR losses for the two operations can be approximated as follows:

$$\frac{\text{SNR}_{\text{new}}}{\text{SNR}_{\text{old}}} = \left[\frac{\sin(D\hat{w}_r 0.00025)}{D\hat{w}_r 0.00025}\right]^2 \left[\frac{\sin(LD\hat{w}_f 0.00025)}{LD\hat{w}_f 0.00025}\right]^2 \quad (12)$$

where $D\hat{w}_r$ is the spacing of the rough grid and $D\hat{w}_f$ is the spacing of the fine grid, both in rad/sec. Thus, if one is using the alternate half-bits method, for which L = 10, and if one uses $D\hat{w}_r/(2p)$ = 250 Hz and $D\hat{w}_f/(2p)$ = 25 Hz, then the worst-case pre-squaring SNR loss will be 0.45 dB. If one doubles both grid spacings, then the worst-case pre-squaring SNR loss becomes 1.82 dB. The importance of this loss doubles after the squaring operation. Therefore, one must not use Doppler shift grids that are too coarse if one wants to detect very weak signals.

Reduction of the Number of FFT Operations via Frequency Shifting

The acquisition algorithms call for calculation of the FFT of $y_k \exp[-j(w_{IF} - \hat{w}_D)t_k]$ at each frequency point on the Doppler shift rough grid, as in eq. (5). Many of these FFT operations can be avoided by performing shifts in the frequency domain. If one knows the FFT of $y_k \exp[-j(w_{IF} - \hat{w}_D)t_k]$ for k = 0,…, N-1, then the FFT of $y_k \exp[-j(w_{IF} - \hat{w}_D - D\hat{w}_{Dp})t_k]$ can be computed by a simple circular frequency shift

$$\text{FFT}\left(\begin{bmatrix} y_0 \exp[-j(w_{IF} - \hat{w}_D - D\hat{w}_{Dp})t_0] \\ y_1 \exp[-j(w_{IF} - \hat{w}_D - D\hat{w}_{Dp})t_1] \\ \vdots \\ y_{N-1} \exp[-j(w_{IF} - \hat{w}_D - D\hat{w}_{Dp})t_{N-1}] \end{bmatrix}\right) =$$

$$\exp[jD\hat{w}_{Dp}t_0] \begin{bmatrix} Y_{N-p} \\ \vdots \\ Y_{N-1} \\ Y_0 \\ \\ Y_{N-p-1} \end{bmatrix} \quad (13)$$

if $D\hat{w}_{Dp} = p\times 2p\times 1000\times[1 + \hat{w}_D/(2p\times 1575.42\times 10^6)]$ for any integer p. If one chooses the rough Doppler shift grid points to be $p\times 2p\times 1000$ for integer values of p, then one only needs to compute the first FFT in eq. (5) once, at a Doppler shift of zero. (Note that the second FFT in eq. (5) gets computed only once due to the assumption that the sampled PRN code is periodic of period N.) Of course, one still needs to compute the IFFT of eq. (6) at each Doppler shift grid point. Thus, this procedure can save about half of the required FFT/IFFT calculations.

One must modify the correlation interpolation associated with eq. (11) if one uses eq. (13) in order to economize on FFTs. The definition of the $D\hat{h}$ chipping rate perturbation must be based on the frequency difference between the Doppler-shift at the fine grid point and the Doppler shift that has been used in the original FFT calculation, $\hat{w}_D$ of eq. (13).

Normally one wants to use a tighter spacing for the rough Doppler shift grid than 1000 Hz. The first term on the right-hand side of eq. (12) implies that a 1000 Hz spacing will yield a worst-case SNR loss of 3.9 dB. This could be unacceptable for a low SNR signal acquisition.

The way to use eq. (13) with a narrower spacing of the rough grid points is to calculate several eq.-(5)-type FFTs for a minimal set of frequencies on the rough grid that are not 1000 Hz multiples of each other. For example, suppose one used the rough Doppler shift grid points …, -1500 Hz, -1000 Hz, -500 Hz, 0 Hz, 500 Hz, 1000 Hz, 1500 Hz, ... One could calculate all of the required FFTs at the frequencies …, -1000 Hz, 0 Hz, 1000 Hz, … by using frequency shifts of the FFT for 0 Hz, and one could calculate the FFTs at …, -1500 Hz, -500 Hz, 500 Hz, 1500 Hz, …by frequency shifts of the result at 500 Hz. Thus, one would need to calculate only 2 FFTs in order to halve the grid spacing. Note that the actual grid frequencies associated with the 500 Hz FFT will be −1500.0006, -500.0003, 500, 1500.0003 Hz, … The small perturbations in these values result from the definition of $D\hat{w}_{Dp}$ that appears just after eq. (13).

This technique has not been used in the examples of this paper. It can greatly increase the computer storage requirements for long data intervals, and it saves no more than a factor of 2 in computation time. Therefore, it was not deemed to be worth implementing in order to generate this study's proof-of-concept results.

Efficient Calculation of the Acquisition Statistic for the Full-Bit Method

One must be careful about how one calculates P_{long} for the full-bit method. Twenty different values of P_{long} must be calculated for a given $[\hat{t}_{s(n)}, \hat{w}_D]$ combination because there are 20 different possible bit start times. An efficient way to do this is to calculate running I and Q sums that include the last 20 accumulations:

$$I_{run(\mathbf{l})} = \sum_{k=\mathbf{l}-19}^{\mathbf{l}} I_k \quad \text{and} \quad Q_{run(\mathbf{l})} = \sum_{k=\mathbf{l}-19}^{\mathbf{l}} Q_k \quad (14)$$

These sums can be computed efficiently via the recursions

$$I_{run(\mathbf{l}+1)} = I_{run(\mathbf{l})} + I_{\mathbf{l}+1} - I_{\mathbf{l}-19} \quad \text{and}$$
$$Q_{run(\mathbf{l}+1)} = Q_{run(\mathbf{l})} + Q_{\mathbf{l}+1} - Q_{\mathbf{l}-19} \quad (15)$$

These recursions require storage of the last 20 values of $I_{\boldsymbol{I}}$ and $Q_{\boldsymbol{I}}$, presumably in a circularly indexed array. Equations (15) get iterated once per PRN code period. After each iteration $I_{run(\boldsymbol{I})}$ and $Q_{run(\boldsymbol{I})}$ get squared and added to the P_{long} value that corresponds to i = mod($\boldsymbol{I}$+1,20).

WEAK SIGNAL ACQUISITION IN THE PRESENCE OF INTERFERING STRONG SIGNALS

The acquisition of a weak GPS signal is more difficult if there are other GPS signals present whose power is much stronger than that of the signal in question. One may have to use additional strategies in order to detect the signal reliably in this case. The interfering signals have a slight, non-zero cross correlation with the weak signal. This cross correlation can cause a false acquisition if the weak signal has about 23 dB less power than the strong signal. This spoofing can occur even when the two signals' Doppler shifts are very different because the product of 2 PRN codes has harmonic components at multiples of 1000 Hz.

Appendix B of Ref. 6 mentions a way to deal with this problem: raise the acquisition threshold for P_{long}. This conservative technique reduces the incidence of false alarms due to cross correlation, but it also increases the probability of a missed detection. In high-altitude orbit determination, this approach might preclude the use of signals from the side lobes of GPS transmitters when a signal from another satellite's main lobe is present. Such conservatism would have a negative impact on the ability to do GPS-based orbit determination at high altitude.

Another way to deal with this problem is to subtract the interfering signals out of the raw data before trying to acquire the weak signal. Suppose that one has acquired a stronger signal using the techniques given above. One can then track this signal using techniques like those of Ref. 12. This yields a knowledge of all of the strong signal's parameters in eq. (1) except for A, the signal amplitude. Suppose that the signal s_k is defined to be the PRN code and carrier part of the tracked strong signal:

$$s_k = c[(1+h)(t_k - t_s)]\, cos[w_{IF}t_k - (w_D t_k + f_0)] \qquad (16)$$

where t_s, w_D, and f_0 are estimated by the techniques of Ref. 12 and may be time-varying. Suppose, also, that one knows the bit transition times of the strong signal. Then one can estimate the signed amplitude of the strong signal over a given data bit interval as follows:

$$\hat{A} = \sum_{k=bit\,start}^{bit\,stop} s_k y_k \Big/ \sum_{k=bit\,start}^{bit\,stop} s_k s_k \qquad (17)$$

One can then use $\hat{A}$ to subtract the estimated strong signal out of the raw data in order to create w_k, a signal that still contains the weak signal but that does not contain the strong signal:

$$w_k = y_k - \hat{A}\, s_k \qquad (18)$$

Of course, one must re-calculate $\hat{A}$ for every data bit interval. The operations in eqs. (17) and (18) amount to a projection of the signal y_k perpendicular to the signal s_k as measured in signal-time-history space. This particular implementation assumes that the signal amplitude does not vary significantly over a 0.020 sec data bit period.

WEAK SIGNAL ACQUISITION RESULTS

Test Procedures

The signal acquisition algorithms of this paper have been tested using actual data from a GPS bit grabber. The bit-grabber has been connected to a roof-mounted patch antenna that has a low-noise pre-amplifier. The raw data contains a number of strong GPS signals. Simulated noise has been added to the raw bit-grabbed data in order to create data sets that contain GPS signals with low SNRs. These data sets have been used to test the acquisition algorithms.

The bit grabber mixes the signal down to a nominal IF of 4.309 MHz and samples it at 5.714 MHz. It use automatic gain control and 2-bit analog-to-digital conversion (ADC). The sampling process aliases the signal down to 1.405 MHz and creates the phase reversal of eq. (1) [12].

The strong GPS signals in the raw bit-grabbed data have been acquired and tracked in order to determine truth values for t_s and w_D. Acquisition has been accomplished using the algorithms of this paper. Tracking has been accomplished using the algorithms of Ref. 12. The tracking algorithms yield very accurate estimates of t_s and w_D because they use optimal smoothing and because the raw signals are very strong.

Weak signals have been generated by adding simulated colored noise to the raw bit-grabbed data. A model of the RF front end's band-pass filters has been driven by simulated white noise in order to produce simulated colored noise with the proper spectrum. The amplitude of this noise has been adjusted in order to achieve a certain pre-specified loss of SNR, and the noise has been added to the raw RF bit grabbed data. Finally, in operations which are identical to those of the bit grabber, the signal has been gain controlled and re-digitized using a 2-bit quantization.

The SNR of the weakened signal has been checked by using the strong signal truth results. The strong signal results give the carrier phase, the code phase, and the bit transitions. These can be used in to calculate I and Q accumulations, as in eqs. (3a) and (3b), and to estimate the signal amplitude A, as in eq. (17). The measured

standard deviation of the I and Q accumulations, s_{IQ}, and the measured value of A have been used to calculate the carrier to noise ratio:

$$C/N_0 = 10 \log_{10}\{(NA)^2/(4\times 0.001\, s_{IQ}^2)\} \text{ dB Hz} \quad (19)$$

This SNR definition is larger by 3 dB than that used in Appendix B of Ref. 6.

Example Cases

A number of weak signal acquisition cases have been run. Signals with SNRs as low as 20.7 dB Hz have been successfully acquired, and acquisition times as long as 3.980 sec (M = 199 bits) have been used. The alternate half-bits method has been compared with the full-bits method, and the dependence of the required acquisition time on the SNR has been investigated.

The following results are those of a typical case. It uses 4 seconds worth of data for the GPS satellite with PRN code number 08. The data has been collected on Jan. 10, 2001 in Ithaca, N.Y. The SNR of the raw data is 50.7 dB Hz. This SNR has been attenuated by 30 dB through the addition of simulated noise. Thus, the test acquisition has been carried out at an SNR level of 20.7 dB Hz. The full 4 second data batch has been used in the acquisition process because an analysis of the missed detection and false alarm probabilities indicated that this amount of data would virtually ensure success. This means that M = 199 data bits have been used in the full-bits process; the bit optimization process uses data that spans M + 19/20 bits. Both the full-bits process and the alternate half-bits process have been tried on this data set.

The acquisition has searched in the Doppler shift space spanning from –5250 Hz to + 5250 Hz, which is the expected range of signals for a static terrestrial receiver. The fine Doppler shift grid spacing has been set at 12.5 Hz for the full-bits procedure and at 25 Hz for the half-bits procedure. The rough Doppler shift grid spacing has been set at 250 Hz for both methods.

An upper bound for the false alarm probability has been set at 0.001. Recall that this is a bound on the probability that any of the peaks in the full $[t_s, w_D]$ search space will reach the P_{min} threshold when no signal is present. For the 199 full-bits method this yields a detection threshold of $P_{min} = 600L(s_{IQ})^2$. For the 199 alternate half-bits method the threshold is $P_{min} = 582L(s_{IQ})^2$. The corresponding probabilities of missed detection are 5.3×10^{-9} for the full-bits method and 0.11 for the alternate half-bits method, assuming an SNR of 20.7 dB Hz. These greatly differing probabilities of missed detection illustrate the superior performance of the full-bits method over the half-bits method. In either case, though, the likelihood of successful signal detection is high. The main advantage of the full-bits method is that it should be able to acquire a signal at an SNR level of 18 dB Hz with a missed detection probability of only 0.12 when operating on 199 bits. The corresponding alternate half-bits method has a missed detection probability above 0.36 for SNR levels below 20 dB Hz.

The example signal has been successfully detected by both the full-bits method and the alternate half-bits method. In the full-bits method the maximum value of P_{long} has exceeded P_{min} by a significant amount: $\max(P_{long}) - P_{min} = 232L(s_{IQ})^2$. This margin is much lower in the alternate half-bits case, only $53L(s_{IQ})^2$. The missed-detection probability analysis for the full-bits method predicts that $\max(P_{long})$ will exceed this example's computed value 75 % of time, which further confirms the high likelihood of a successful acquisition.

Figure 2 presents a surface plot of $P_{long}/[L(s_{IQ})^2]$ vs. t_s and w_D for this example's full-bits acquisition. The peak at the correct values of t_s and w_D rises well above the noise. This indicates that this acquisition procedure is a powerful method for detecting the 20.7 dB Hz signal.

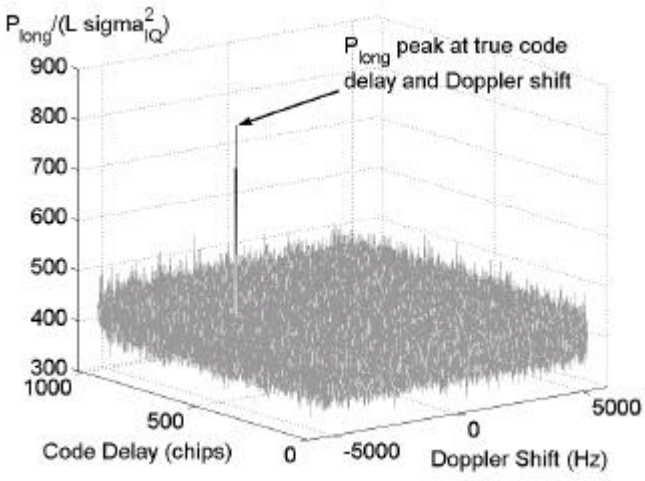

Fig. 2. The signal detection statistic, P_{long}, as a function of the trial values of the code delay and the Doppler shift, a 199 full-bits acquisition case for PRN 08 with an SNR of 20.7 dB Hz.

These results compare favorably with the weak signal acquisition results of Ref. 8. Reference 8 starts out with a strong signal from a simulator. Although the paper never reports an SNR, its Fig. 4 implies that the original signal's SNR is about 50 dB Hz at the output of the RF front end. The paper presents acquisition results for various attenuated versions of the signal. It reports successful acquisition down to 35 dB Hz (15 dB of loss) without assuming knowledge of the GPS data bits, and it claims that one needs to know the GPS data bit transitions *a priori* in order to acquire weaker signals. It reports successful acquisition of signals down to 15 dB

Hz when aided by such *a priori* information. The present example is equivalent to the 30 dB loss case of Ref. 8. Reference 8 reports successful acquisition using only 0.400 sec of data (20 bits) in this case. The current technique uses 10 times as many bits to acquire the signal, but it does so without using *a priori* information about the bit transitions.

The bit transition time has been successfully estimated in this case – almost. In fact, there is no value of i, the PRN code period of the bit transition, that is exactly correct. The peak of the P_{long} vs. t_s plot does not occur at an end of the *a priori* accumulation interval of eqs. (3a) and (3b). Figure 2 indicates that the code start time occurs right in the middle of the accumulation interval. This is the worst possible case for correct i calculation because it causes an i ambiguity. The calculated i for this case has been affected by this ambiguity: Its optimal value is off by one PRN code period from the value that would bring the solved bit transition time as close as possible to the true bit transition time. An unambiguous correct determination of i has been made in another case in which t_s is much closer to the end of the *a priori* accumulation interval. In cases that are ambiguous, it is advisable to re-estimate i at the end of the acquisition using the correct t_s and w_D and using re-aligned accumulation intervals that put t_s at one end.

A Case with a Weak Signal in the Presence of Strong Signals

The case of a weak GPS signal with strong interfering GPS signals has been tried using raw bit-grabber data with no added noise. The weak signal in this case has an SNR of 38 dB Hz, and there are 9 stronger interfering signals with SNRs ranging from 51.6 dB Hz down to 41.9 dB Hz. Acquisition calculations have been attempted both with and without the cancellation of the strong interfering signals, as described in the fifth section. In the cases when strong signals have been cancelled, all 9 have been eliminated from the signal prior to acquisition.

The signal has been successfully acquired, both with the alternate half-bits method and with the full-bits method. It can be acquired using a single half bit (M = 1) and without cancellation of the strong GPS signals, but the margins for avoiding false alarms and missed detections are low in this case. The reliability of the acquisition increases if the number of half-bits M increases, or if the strong signals are cancelled before acquisition.

The beneficial effects of canceling the strong signals increase as M increases. The theory of the acquisition calculation predicts that one can reduce both the probability of false alarm and the probability of a missed detection by increasing M. This calculation, however, is based on the assumption that the noise is random. The presence of stronger interfering signals increases the probability of a false alarm, and the amount of the increase grows with M.

If one cancels the strong signals, then the full benefits of an increased M can be realized. This effect is illustrated by the probability density functions of Fig. 3. This figure plots four probability density functions for the signal detection statistic $P_{long}/[L(s_{IQ})^2]$ under the assumption that no signal is present. These are the density functions that get used to calculate false alarm probabilities. Each corresponds to an alternate half-bits acquisition that uses M = 5 half bits. The black dash-dotted curve is an experimentally determined curve for the case in which strong signals are not cancelled. It has been generated from a histogram of all the P_{long} values corresponding to erroneous values of t_s and w_D. The solid gray curve is the theoretical 10^{th} degree chi-squared distribution that has the same mean as the experimental curve. These curves should be identical, but they differ because of the non-random interference from the strong signals. Significantly, the experimental curve has a heavier tail in the $+\infty$ direction. This implies that the true false alarm probability will be higher than the theory predicts. This discrepancy between the tails of the theoretical plot and the experimental plot tends to increase as M increases.

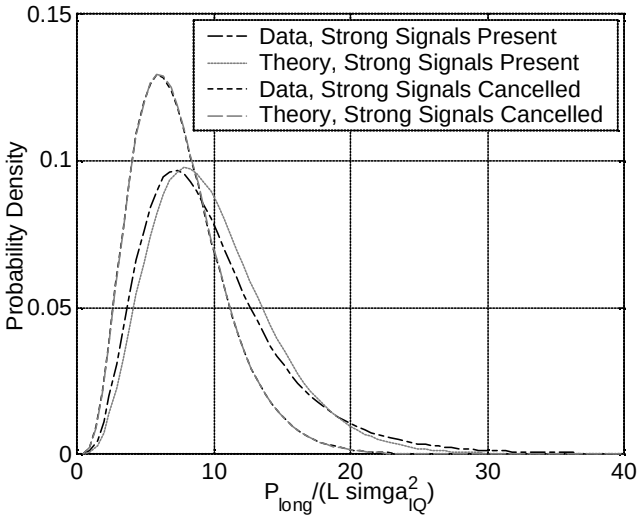

Fig. 3. The effect of strong signal cancellation on the null-hypothesis probability distribution of the detection statistic, a case with 5 alternate half-bits.

The other two curves on the plot correspond to the same data, but with the strong signals cancelled out. The black dotted curve is the experimental curve for this case, and the dashed gray curve is the corresponding theoretical chi-squared distribution. These two curves fall right on top of each other. Thus, the theory does a good job of predicting the experiment if one subtracts out interfering

signals. Also, both of these curves show a no-signal distribution that is shifted markedly towards zero in comparison to the distribution with strong signals present. Thus, there is a greater ability to detect weak signals with both a low probability of false alarm and a low probability of missed detection if one first cancels the strong signals out of the bit-grabbed data.

An interesting side note to this analysis has to do with the effect of the GPS signals on the RF power spectrum near the L1 frequency. Reference 13 points out that the power spectral density of any given GPS C/A signal is about 15 dB below the power spectral density of the noise. This fact might lead one to think that the signals have no noticeable effect on the spectrum. This is not the case. There is a very noticeable effect on the signal spectrum after all 9 of the strongest GPS signals have been cancelled out of the bit-grabbed data.

Trends in Acquisition Times and Detectable Signal-to-Noise Ratios

The required acquisition time is a function of the SNR at the output of the bit grabber. As the SNR decreases the acquisition time increases. Because of the squaring loss, one would expect the acquisition time to go as one over the square of the SNR. Thus, a 5 dB decrease in the received SNR would necessitate a factor of 10 increase in the number of bits or half bits used in the acquisition. This rule of thumb has held roughly true. A 26 dB Hz signal could be acquired using the alternate half-bits algorithm with 20 bits of averaging, but a 21 dB Hz signal required 199 half bits to achieve a successful detection. A full-bits algorithm that used M = 35 bits easily detected a signal whose SNR was 26 dB Hz, but M = 35 was useless on a 21 dB Hz signal. For higher SNR levels this relationship may be more rough, but it is a good rule of thumb for estimating the necessary input parameters of a signal acquisition algorithm.

Analysis indicates that the weakest signal which can be detected in 4 sec of data is one with an SNR of 18 dB Hz. This presumes use of the full-bits method. If one increases the acquisition time to 30 sec, then the weakest detectable signal has an SNR of 14 dB Hz.

Computational Effort

The algorithms described in this paper can be very expensive in terms of computation time and memory. The detection associated with Fig. 2 is very expensive. It requires 36 hours to execute in MATLAB on a 733 MHz Pentium III processor (i.e., 10^{12} floating point operations). The code needs to store about 430 Mbytes of data during its execution. The corresponding alternate-half-bits run requires only 14 hours of execution time and 240 Mbytes of memory. Such expensive computations will be feasible only for off-line applications.

The algorithms of this paper have been partially optimized for speed by careful use of FFT block processing, but there is room for more optimization. The most obvious thing to do is to switch from interpretive MATLAB to compiled code. This should yield a speed up of a factor of 10 to 20 based on the raw machine speed. Another possible source of speed increase would be to use integer calculations in place of double precision real calculations.

The computation time varies linearly with the number of PRN code periods that get processed, $M \times L$, and roughly linearly with the width of the initial Doppler shift uncertainty $w_{Dmax} - w_{Dmin}$. The former quantity is dictated by the received SNR, but the latter quantity can be greatly reduced if one has a rough *a priori* estimate of the user vehicle state. The Fig.-2 calculations use $(w_{Dmax} - w_{Dmin})/(2p)$ = 10,500 Hz. If the initial Doppler uncertainty is reduced to 25 Hz, then the execution time reduces by a factor of 127, to 17 min. Thus, with the proper optimizations, the algorithms of this paper would be practical for use in spacecraft orbit determination via post processing of flight data.

VII. Summary and Conclusions

This paper has developed methods for the acquisition of weak GPS C/A signals without *a priori* information about the GPS data bit stream. These methods are applicable for use in a software receiver. Weak signals are acquired by a process of accumulation of correlations, summation of accumulations, squaring to remove GPS data bits, and further summation of the squared accumulations. The resulting detection statistic is then maximized via a search over bins in the space of possible code delays and Doppler shifts. Two strategies are introduced to avoid the use of pre-squaring summation intervals that contain GPS data bit transitions. One strategy uses only alternate half-bit intervals and does two independent acquisition calculations for two interleaved sets of half-bit intervals. The other strategy estimates the bit transition times by choosing the bit-start PRN code period index that maximizes the detection statistic. The calculations of these detection procedures have been specially tailored to the block processing context, which uses FFT and inverse FFT operations in order to rapidly calculate correlations for entire batches of candidate PRN code period start times.

Additional work has been done on the problem of weak signal detection in the presence of interfering strong GPS signals. The resulting algorithm acquires, tracks, and cancels the strong signals prior to acquiring the weak signals. This allows for a reduced conservatism in the calculation of false alarm rates for the detection of the

weak signal.

The new signal acquisition algorithms have been tested using real data from a GPS bit grabber in conjunction with simulated noise. The digitally-generated noise has been used to simulate a lowering of the received SNR. Signals with signal-to-noise ratios as weak as 21 dB Hz have been successfully detected in 4 sec acquisition intervals. The theory indicates that these algorithms could detect 18 dB Hz signals in the same amount of time and that longer data intervals could be used to detect even weaker signals.

Acknowledgments

This work has been supported in part by NASA under cooperative agreement number NCC5-563. Prof. Marty Wells, a statistician in Cornell's School of Industrial and Labor Relations, provided valuable assistance in understanding the statistical hypothesis testing aspects of this work. Mark Krangle and Warren Scott, Cornell Master of Engineering students, collected the bit-grabbed data that has been used in this study.

References

1. Moreau, M., Axelrad, P., Garrison, J.L., Kelbel, D., and Long, A., "GPS Receiver Architecture and Expected Performance for Autonomous GPS Navigation in Highly Eccentric Orbits," *Proceedings of the ION 55th Annual Meeting*, June 28-30, 1999, Cambridge, MA, pp. 653-665.

2. Balbach, O., Eissfeller, B., Hein, G.W., Enderle, W., Schmidhuber, M., and Lemke, N., "First Results of the GPS Experiment on the HEO Mission Equator-S," *Proceedings of the IEEE 1998 Position, Location, and Navigation Symposium*, April 20-23, 1998, Palm Springs, CA, pp. 243-249.

3. 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.

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

5. Srinivasan, J., Bar-Sever, Y., Bertiger, W., Lichten, S., Muellerschoen, R., Munson, T., Spitzmesser, D., Tien, J., Wu, S.C., and Young, L., "microGPS: On-Orbit Demonstration of a New Approach to GPS for Space Applications," *Proceedings of the ION GPS '98*, Sept. 15-18, 1998, Nashville, TN, pp. 1537-1545.

6. 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.

7. Ward, P.W., "GPS Receiver Search Techniques," *Proceedings of the IEEE 1996 Position, Location, and Navigation Symposium*, April 22-26, 1996, Atlanta, GA, pp. 604-611.

8. Akos, D.M., Normark, P.L., Lee, J.T., Gromov, K.G., Tsui, J.B.Y., and Schamus, J., "Low Power Global Navigation Satellite System (GNSS) Signal Detection and Processing," *Proceedings of the ION GPS 2000*, Sept. 19-22, 2000, Salt Lake City, UT, pp. 784-791.

9. Spilker, J.J. Jr., *Digital Communications by Satellite*, Prentice-Hall, (Englewood Cliffs, N.J., 1977), pp. 446-447.

10. Poor, H.V., *An Introduction to Signal Detection and Estimation*, Springer-Verlag, (New York, 1988), pp. 7-195.

11. Arnold, B.C., and Balakrishnan, N., *Relations, Bounds and Approximations for Order Statistics*, Springer-Verlag, (New York, 1989), pp. 54-55.

12. Psiaki, M.L., "Smoother-Based GPS Signal Tracking in a Software Receiver", submitted to the IEEE Transactions on Aerospace and Electronics Systems, in review. Currently available at http:// www.mae.cornell.edu/Psiaki/gpssigsmoother.pdf.

13. Spilker, J.J. Jr., "GPS Signal Structure and Theoretical Performance," 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. 88-89.