Extended Kalman Filter Methods for Tracking Weak GPS Signals

Mark L. Psiaki and Hee Jung, *Cornell University, Ithaca, N.Y.*

BIOGRAPHIES

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 GPS technology and applications.

Hee Jung is a Ph.D. candidate in Mechanical and Aerospace Engineering at Cornell University. She received her BS and MS in Astronomy from the Seoul National University in Korea and another MS in Aerospace Engineering from Texas A&M University. Her main research interests are orbit and attitude determination of satellites with GPS applications.

ABSTRACT

A combined phase-locked loop/delay-locked loop has been developed for tracking weak GPS C/A signals. This work enables the use of the weak side-lobe signals that are available at geosynchronous altitudes. The tracking algorithm is an extended Kalman filter (EKF) that estimates code phase, carrier phase, Doppler shift, rate of change of Doppler shift, carrier amplitude and data bit sign. It forms a likelihood function that depends on the errors between accumulations and their predicted values. It recursively minimizes this likelihood function in order to track the signal. It deals with data bit uncertainty using a Bayesian analysis that determines a posteriori probabilities for each bit sign. A second filter is used to initialize the EKF. This batch filter starts with coarse carrier frequency and code phase estimates and refines them using maximum likelihood techniques while estimating the carrier phase and the first PRN code period of a navigation data bit. The resulting system can acquire and maintain lock on a signal as weak as 15 dB Hz if the receiver clock is an ovenized crystal oscillator and if the line-of-sight acceleration variations are as mild as those seen by a geostationary user vehicle.

I. INTRODUCTION

Tracking algorithms allow a GPS receiver to maintain lock on the Doppler shift and the pseudo-random number (PRN) spreading code of a received signal so that the receiver can determine navigation observables and decode the navigation data message. One necessary tracking algorithm is a delay-locked loop (DLL), which maintains phase alignment between the received PRN code and a replica in the receiver. A receiver uses either a frequency-locked loop (FLL) or a phase-locked loop to align a replica of the carrier signal with the received carrier. Various designs exist for these tracking loops, e.g., see Refs. 1 and 2.

The goal of the present work is to design a combined DLL/PLL code and carrier tracking loop that is effective at tracking GPS L1 Coarse/Acquisition signals (C/A) for very low carrier-to-noise ratios. This is to be accomplished without prior knowledge of the navigation data bits. These tracking loops will be developed using optimal estimation techniques. Standard DLLs and PLLs, while robust, are not optimal. PLLs lose lock at low carrier-to-noise ratios because of the nonlinearities in their discriminators and because of dynamic variations of the signal's phase [2]. Uncertainty about the navigation data bits greatly exacerbates the problems of PLLs at low carrier-to-noise ratios because bit error rates become large and lead to a destabilizing feedback effect. The goal of using optimal techniques is to deal with these problems in the best possible way and thereby lower the carrier-to-noise threshold at which loss of lock occurs for a given level of signal phase dynamics.

The motivation for this work comes from a desire to use GPS for high-altitude spacecraft navigation, above the constellation. Typical off-the-shelf receivers can track signals down to a carrier-to-noise density, C/N_0, of about 35 dB Hz, which is marginally sufficient for tracking main-lobe signals at geosynchronous altitudes using a patch antenna. Simulation studies of high-altitude navigation performance predict that significant gains can be made if signals can be used effectively down to 28 dB Hz [3,4]. This present work aims to track signals in the 12-29 dB Hz range. This will enable the use of side-lobe signals received at geosynchronous altitudes using a patch antenna [5]. Additional motivation for this work comes from a desire to track transient weak signals that occur terrestrially during ionospheric scintillations [6]. Such an ability would enable physicists to extract a

greater amount of information from GPS soundings of the disturbed ionosphere.

The present work makes three main contributions. First, it develops a new fine acquisition algorithm. This procedure is needed in order to provide the tracking algorithm with accurate initial estimates of the carrier frequency, carrier Doppler shift, code phase, and carrier amplitude. It starts with coarse estimates of the carrier Doppler shift and the code phase and uses a batch nonlinear filter to refine these estimates while solving for initial carrier phase and carrier amplitude. The paper's second contribution is the development of a combined carrier- and code-tracking nonlinear Kalman filter. A Kalman filter is an optimal algorithm that is efficient for real-time implementation because of its iterative-in-time nature. The third contribution is a Bayesian adaptation of nonlinear Kalman filtering techniques that deals effectively with the uncertain navigation data bits when the carrier-to-noise density is low. The new fine acquisition and tracking algorithms are tested using a high-fidelity simulation of the weak GPS signal that exits a receiver's RF front end.

Some of these algorithms require a special receiver architecture. The coarse and fine acquisitions must take place in a software receiver environment because the available algorithms for coarse acquisition of weak signals [7] and the new algorithm for fine acquisition are batch algorithms. The new tracking algorithm operates in an iterative causal manner using standard accumulations. It can be implemented in a software receiver or in a standard real-time receiver that uses a hardware correlator. This paper envisions a system that acquires an initial batch of data for coarse and fine acquisition. While it is processing this data, it continues to record a stream of intermediate frequency data from its RF front end for later use in tracking. After it has finished its coarse and fine acquisition calculations, it initiates the new weak signal tracking algorithm. It starts out tracking stored data and operates faster than real-time in order to "catch up". Afterwards it continues to track a given signal in real time. Systems that lack sufficient power to do batch-mode acquisition, interim RF bit storage, and "catch-up" tracking still could use the new tracking algorithm. Its weak signal capabilities would be useful for a signal that suffered a power fade subsequent to having been acquired by standard techniques.

The remainder of this paper includes 4 main sections followed by conclusions. Section II presents models of the correlation measurements and the signal. Section III describes the fine acquisition algorithm. Section IV explains the Kalman filter-based tracking algorithm. Section V presents simulation results for these algorithms. Section VI gives the conclusions.

II. MODELS OF THE CORRELATION MEASUREMENTS AND THE SIGNAL DYNAMICS

The batch fine-acquisition algorithm and the signal tracking Kalman filter operate using dynamic models of carrier phase, code phase, and carrier amplitude and measurement models that give the relationship between these signal quantities and the correlations that the receiver uses to sense the signal. These models assume that the sampled signal coming out of the receiver's RF front end takes the form

$$y_j = A(t_j)d[t_j - t_s(t_j)]\, C[t_j - t_s(t_j)]\, \cos[w_{IF}\, t_j - f(t_j)] + n_j \qquad (1)$$

where y_j is the measured RF front-end output at sample time t_j, $A(t)$ is the carrier amplitude, $d(t)$ is the 50 Hz navigation data bit stream of ±1 values, $C(t)$ is the 1.023 MHz C/A PRN bit stream of ±1 values, $t_s(t)$ is the PRN code phase expressed as a relative code start time, w_{IF} is the RF front-end's intermediate image of the nominal GPS L1 carrier frequency, $f(t)$ is the carrier phase that results from accumulated delta range, and n_j is an element of a zero-mean discrete-time Gaussian white noise sequence with a variance of s_n^2. The carrier-to-noise density of this sampled signal is $C/N_0 = A^2/(4 s_n^2 dt)$, where $dt = t_{j+1} - t_j$.

Accumulation Measurement Models. The receiver accumulates correlations between the y_j data stream and replicas of the code and carrier signals that it produces. These accumulations take the usual forms

$$I_k(D) = \sum_{j=j_k+1}^{j_k+N_k} y_j C_{NCO}(t_j + D - t_{NCOk}) \cos[w_{IF}\, t_j - f_{NCO}(t_j)] \qquad (2a)$$

$$Q_k(D) = -\sum_{j=j_k+1}^{j_k+N_k} y_j C_{NCO}(t_j + D - t_{NCOk}) \sin[w_{IF}\, t_j - f_{NCO}(t_j)] \qquad (2b)$$

where $I_k(D)$ and $Q_k(D)$ are, respectively, the in-phase and quadrature accumulations with an "early" offset of D, which will be a late offset if $D < 0$. The function $C_{NCO}(t)$ is the receiver's reproduction of the tracked PRN code. The subscript NCO is used because this function simulates a code numerically controlled oscillator. The time t_{NCOk} is both the NCO's prompt code start time and the start time of the accumulation interval. The initial sample j_k+1 is the minimum value that respects the bound $t_{NCOk} \le t_{jk+1}$. Each accumulation spans one PRN code period; therefore, the number of samples N_k is the maximum value that respects the limit $t_{jk+Nk} < t_{NCOk+1}$. The function $f_{NCO}(t)$ is the receiver's reproduction of the signal's carrier phase. The negative sign in front of it presumes high-side mixing during one of the RF front-end's down conversion stages.

Equations (2a) and (2b) give a recipe for how the

receiver calculates its accumulations, but the fine acquisition algorithm and the Kalman filter need a model of how these accumulations are related to the actual signal parameters in eq. (1). This model takes the form:

$$I_k(D) = \frac{N_k \bar{A}_k d_m}{2} \cos(Df_k) R(Dt_k + D) + n_{Ik} \quad (3a)$$

$$Q_k(D) = -\frac{N_k \bar{A}_k d_m}{2} \sin(Df_k) R(Dt_k + D) + n_{Qk} \quad (3b)$$

where $\bar{A}_k$ is the average carrier amplitude over the accumulation interval, d_m is the navigation data bit, Df_k is the interval average of the carrier phase error $f(t)-f_{NCO}(t)$, $Dt_k = t_s(t_{midk})-t_{midk}$ is the code phase error at the interval mid-point $t_{midk} = (t_{NCOk}+t_{NCOk+1})/2$, $R(t)$ is the autocorrelation function of the PRN code, and n_{Ik} and n_{Qk} are samples of zero-mean, uncorrelated Gaussian discrete-time white noise sequences, both with variance equal to $N_k s_n^2/2$. The model of $R(t)$ that gets used has the slope discontinuities of its triangular peak rounded off by cubic splines. These cause d^2R/dt^2 to be continuous, which avoids problems in the gradient-based numerical optimizations that are part of the fine acquisition algorithm and the Kalman filter. This modification of $R(t)$ is reasonable because the limited bandwidth of the RF front end rounds off the actual correlation's sharp corners.

The Kalman filter make use of prompt and early-minus-late accumulations that are summed over entire navigation data bit periods. These form the 4×1 measurement vector

$$y_m = \frac{1}{s_n}\sqrt{\frac{2}{N_m}}\begin{bmatrix} \sum_{k=k_m}^{k_m+19} I_k(0) \\ \sum_{k=k_m}^{k_m+19} Q_k(0) \\ \frac{1}{\sqrt{h_{eml}}} \sum_{k=k_m}^{k_m+19} [I_k(D_{eml}/2) - I_k(-D_{eml}/2)] \\ \frac{1}{\sqrt{h_{eml}}} \sum_{k=k_m}^{k_m+19} [Q_k(D_{eml}/2) - Q_k(-D_{eml}/2)] \end{bmatrix} \quad (4)$$

where m is the index of the navigation data bit interval, $N_m = (N_{km} + N_{km+1} + \ldots + N_{km+19})$ is the number of RF front-end samples in the data bit interval, k_m is the index of the first PRN code period of the data bit interval, D_{eml} is the time offset between the early and late versions of the receiver's PRN code reconstruction, and $h_{eml} = 2[1-R(D_{eml})]$ is a constant. Equation (4) gives a recipe for how to compute y_m from receiver data, but the Kalman filter also requires a model of how y_m is related to signal parameters. It takes the form:

$$y_m = \frac{\bar{A}_m d_m}{s_n}\sqrt{\frac{N_m}{2}}\begin{bmatrix} \cos(Df_m)R(Dt_m) \\ -\sin(Df_m)R(Dt_m) \\ \frac{1}{\sqrt{h_{eml}}}\cos(Df_m)R_{eml}(Dt_m) \\ \frac{-1}{\sqrt{h_{eml}}}\sin(Df_m)R_{eml}(Dt_m) \end{bmatrix} + n_{ym}$$

$$= d_m\, h_m(Df_m, Dt_m, \bar{A}_m) + n_{ym} \quad (5)$$

where $\bar{A}_m$ is the average carrier amplitude over the bit interval, Df_m is the average carrier phase error over the bit interval, $Dt_m = t_s(t_{midm})-t_{midm}$ is the code phase error at the mid-point of the bit interval $t_{midm} = (t_{NCOkm}+t_{NCOkm+20})/2$, $R_{eml}(Dt) = R(Dt+D_{eml}/2)-R(Dt-D_{eml}/2)$ is the early-minus-late correlation function, and n_{ym} is a sample from a zero-mean, uncorrelated Gaussian discrete-time white noise vector sequence with covariance equal to the 4×4 identity matrix. Equations (4) has been specifically designed in order to achieve this normalization of the n_{ym} measurement error covariance.

When performing optimal estimation it is best to model the raw measurements and their errors statistically and to use those models directly in the design of the estimator. One should avoid ad-hoc processing of measured data prior to statistical analysis, which is why discriminators are not used here. This paper's approach, however, is in slight violation of this principle. The rawest form of the measurements is given in eq. (1). The new algorithms do not work directly with eq. (1) because their optimization-based techniques would require many re-calculations of correlations, which would be inefficient. Also, the PRN code function $C(t)$ is not everywhere differentiable, which would cause problems for these gradient-based optimization procedures. This "impure" approach does not cause significant loss of performance because of the following facts: The measurements in eqs. (2a), (2b), and (4) capture almost all of the important information about the signal, the measurement models in eqs. (3a), (3b), and (5) are faithful, and the noise in these measurements remains Gaussian.

Carrier Phase, Code Phase, and Carrier Amplitude Dynamic Models. The carrier phase dynamics model takes the form of a discrete-time triple integrator driven by discrete-time white noise:

$$\begin{bmatrix} x_f \\ x_w \\ x_a \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & dt_k & \frac{dt_k^2}{2} \\ 0 & 1 & dt_k \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_f \\ x_w \\ x_a \end{bmatrix}_k - \begin{bmatrix} dt_k \\ 0 \\ 0 \end{bmatrix} w_{NCOk} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} w_{fk} \quad (6)$$

where $x_{fk} = f(t_{NCOk}) - f_{NCO}(t_{NCOk})$ is the difference between the true carrier phase and the receiver NCO's carrier phase at the start of an accumulation interval, x_{wk} is the true carrier Doppler shift at the start of the interval, x_{ak} is the rate of change of the carrier Doppler shift at the start of the interval, w_{NCOk} is the receiver NCO's reconstructed carrier Doppler shift for the interval, $dt_k = t_{NCOk+1} - t_{NCOk}$ is the length of the interval, and w_{fk} is a member of a zero-mean, discrete time Gaussian white-noise vector sequence. The states of this model can be used to deduce the average carrier phase error over the accumulation interval:

$$Df_k = \begin{bmatrix} 1 & \frac{dt_k}{2} & \frac{dt_k^2}{6} \end{bmatrix} \begin{bmatrix} x_f \\ x_w \\ x_a \end{bmatrix}_k - \frac{dt_k}{2} w_{NCOk} + \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} w_{fk} \tag{7}$$

This quantity is needed in eqs. (3a) and (3b). A similar carrier phase model applies when accumulations get summed over bit intervals, as in eq. (4). In this case, the k index of the PRN code period in eqs. (6) and (7) changes to the m index of the data bit period, and the nominal accumulation interval increases from dt_k = 0.001 sec to dt_m = 0.020 sec.

The covariance of the w_{fk} white-noise sequence is a combination of three terms. One models a random walk acceleration of the line-of-sight (LOS) vector to the GPS satellite, as in Ref. 8, and the other two model the effects of receiver clock phase random walk and frequency random walk, as in Ref. 9. The w_{fk} covariance takes the form:

$$\begin{aligned} P_{wfwfk} = E\{ w_{fk} w_{fk}^T \} = \\ q_{LOS} \begin{bmatrix} dt_k^5/20 & dt_k^4/8 & dt_k^3/6 & dt_k^5/72 \\ dt_k^4/8 & dt_k^3/3 & dt_k^2/2 & dt_k^4/30 \\ dt_k^3/6 & dt_k^2/2 & dt_k & dt_k^3/24 \\ dt_k^5/72 & dt_k^4/30 & dt_k^3/24 & dt_k^5/252 \end{bmatrix} \\ + S_g w_{L1}^2 \begin{bmatrix} dt_k^3/3 & dt_k^2/2 & 0 & dt_k^3/8 \\ dt_k^2/2 & dt_k & 0 & dt_k^2/6 \\ 0 & 0 & 0 & 0 \\ dt_k^3/8 & dt_k^2/6 & 0 & dt_k^3/20 \end{bmatrix} \\ + S_f w_{L1}^2 \begin{bmatrix} dt_k & 0 & 0 & dt_k/2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ dt_k/2 & 0 & 0 & dt_k/3 \end{bmatrix} \end{aligned} \tag{8}$$

The quantity q_{LOS} is the intensity of the acceleration random walk. S_g and S_f are the receiver clock's frequency and phase random walk intensities, respectively [9], and w_{L1} is the nominal L1 carrier frequency.

The dynamic model of the PRN code phase propagates the code start time from the beginning of one code period to the next:

$$t_{sk+1} = t_{sk} + \frac{w_{L1} dt_{nom} - \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} w_{fk}}{w_{L1} + x_{wk} + 0.5 dt_{nom} x_{ak}} + w_{tsk} \tag{9}$$

where t_{sk} and t_{sk+1} are the true start and stop times of the PRN code period in question. This model includes carrier aiding via the second term on the right-hand side of eq. (9). The time increment dt_{nom} is the nominal code period, 0.001 sec. The scalar w_{tsk} is a white-noise sequence that effectively models code-carrier divergence as a random walk. Its variance is $E\{w_{tsk}^2\} = dt_k q_{ts}$, where q_{ts} is the random walk intensity. The PRN code start/stop times in eq. (9) can be used to calculate the code phase error Dt_k that is used in eq. (3a) and (3b):

$$\begin{aligned} Dt_k &= (t_{sk+1} + t_{sk})/2 - t_{midk} \\ &= t_{sk} + \frac{1}{2} \frac{w_{L1} dt_{nom} - \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} w_{fk}}{w_{L1} + x_{wk} + 0.5 dt_{nom} x_{ak}} + \frac{1}{2} w_{tsk} - t_{midk} \end{aligned} \tag{10}$$

A model similar to eqs. (9) and (10) applies for propagation of the navigation data bit start/stop times from one bit to the next. In this case, the time index k gets replaced by the time index m, and the interval dt_{nom} increases to 0.020 sec.

A random walk is used to model the dynamics of the carrier amplitude:

$$A_{k+1} = A_k + w_{Ak} \tag{11}$$

A_k and A_{k+1} are the carrier amplitudes at the times t_{NCOk} and t_{NCOk+1}. The scalar w_{Ak} is the zero-mean, white-noise Gaussian sequence that drives the random walk. Its variance is $E\{w_{Ak}^2\} = dt_k q_A$, where q_A is the random walk intensity. The amplitudes in eq. (11) can be used to compute the average amplitude that is needed in eqs. (3a) and (3b):

$$\bar{A}_k = (A_{k+1} + A_k)/2 = A_k + 0.5 w_{Ak} \tag{12}$$

Equations (11) and (12) can be modified to propagate the carrier amplitude from one navigation data bit start time to the next. The only necessary change is to switch from the PRN code period index k to the data bit index m.

III. BATCH FINE ACQUISITION ALGORITHM

The Kalman filter tracking algorithm of this paper needs an accurate initialization in order to function properly in a weak signal environment. The initialization procedure starts with the usual search in code-phase/carrier-Doppler space. For purposes of the present paper, this search is called the coarse acquisition. A suitable batch coarse acquisition algorithm that works well in a weak signal environment is described in Ref. 7. This algorithm gives Doppler shift to an accuracy of about ± 6 Hz and code phase to an accuracy of ± 0.1 chips.

The Kalman filter tracker needs the initial Doppler shift to an accuracy about $0.25/t_{PLL}$ Hz, where t_{PLL} is the characteristic settling time of the filter. This time can be on the order of 1 sec when tracking a weak signal with slow dynamics. In addition, the Kalman filter needs to know the PRN code period that corresponds to the start time of a navigation data bit, it needs an initial carrier phase estimate that is accurate to within $\pm 45^{o}$, and it needs initial estimates of the carrier amplitude and the rate of change of Doppler shift. The Kalman filter also needs a covariance matrix for the initial estimation errors in its 5 states.

A sequence of fine acquisition calculations is used to determine suitable values for the Kalman filter initialization. The first calculation determines the navigation data bit start time. The second calculation estimates initial values for the three carrier phase states, x_f, x_w, and x_a, and it uses these estimates to get a first estimate of the carrier amplitude, A. The final calculation refines the estimates of these 4 quantities along with the estimate of the initial code phase, t_s.

Open-Loop Accumulation Data Used in Fine Acquisition Calculations. The first step in the fine acquisition is to calculate an "open-loop" time history of 1000 Hz accumulations. This computation uses the Doppler shift and code phase estimates from the coarse acquisition, w_{NCO} and t_{NCO0}, to specify an "open-loop" carrier phase time history via the formula $f_{NCO}(t) = w_{NCO}t$ and an "open-loop" code phase time history via iteration of the difference equation $t_{NCOk+1} = t_{NCOk} + 0.001w_{L1}/(w_{L1}+w_{NCO})$. These phase time histories are used to generate in-phase and quadrature prompt and early-minus-late accumulations using the formulas in eqs. (2a) and (2b). Let the prompt accumulations be $I_{pk} = I_k(0)$ and $Q_{pk} = Q_k(0)$, and let the early-minus-late accumulations be $I_{emlk} = I_k(D_{eml}/2) - I_k(-D_{eml}/2)$ and $Q_{emlk} = Q_k(D_{eml}/2) - Q_k(-D_{eml}/2)$. These quantities constitute the data that get used throughout the remainder of the fine acquisition calculations. They are calculated for $k = 0, \ldots, N_{acq}$. N_{acq} is normally chosen to equal the number of PRN code periods used during the coarse acquisition.

Determination of the Start Time of a Navigation Data Bit. The objective of the bit start time calculation is to determine the value of k_0, the PRN code period index whose start time is also the start time of the first navigation data bit. The true value k_0 is an element of the set $\{0,1,2,\ldots,19\}$. It can be determined by an integer optimization [1].

The merit function that gets optimized is the signal power after summing over a pre-detection interval of 1 navigation data bit:

$$J(k_0) = \sum_{m=0}^{M} \left[\left(\frac{1}{N_m} \sum_{k=k_0+20m}^{k_0+20m+19} I_{pk} \right)^2 + \left(\frac{1}{N_m} \sum_{k=k_0+20m}^{k_0+20m+19} Q_{pk} \right)^2 \right] \quad (13)$$

where M is the maximum integer that respects the bound $M \le (N_{acq}-38)/20$. $M+1$ is the number of full data bit intervals. The integer N_m is the total number of RF front-end samples during a proposed navigation data bit interval, as in eq. (4). It normalizes the different summations.

The bit start-stop determination is a brute-force optimization of $J(k_0)$. All 20 possible values of k_0 are tried, and the value that yields the largest $J(k_0)$ gets chosen as the correct bit start/stop index. If M is large enough, then this method is very likely to yield the correct data bit start/stop time. If M is not very large, then one can increase M after the acquisition by using the accumulations from the subsequent tracking interval. This approach further increases the likelihood of getting the correct value of k_0, which is needed to calculate code pseudorange.

Batch Fine Acquisition Cost Functions. The batch acquisition calculations make use of several different cost functions that are related to the joint probability density function of the noise terms in eqs. (3a) and (3b) or (5). The ultimate cost function that the initial estimates x_{f0}, x_{w0}, x_{a0}, A_0, and t_{s0} must minimize is

$$\begin{aligned} J_a(x_{f0},x_{w0},x_{a0},A_0,t_{s0}) = \\ & \frac{1}{s_n^2} \sum_{m=0}^{M} \left\{ \sum_{k=k_0+20m}^{k_0+20m+19} \frac{1}{N_k} \left[I_{pk}^2 + Q_{pk}^2 + \frac{N_k^2 A_0^2}{4} R^2(Dt_k) \right] \right\} \\ & - \sum_{m=0}^{M} ln\left(2\cosh\left\{ \frac{1}{s_n^2} \sum_{k=k_0+20m}^{k_0+20m+19} \left[A_0 R(Dt_k) \right. \right. \right. \\ & \qquad \left. \left. \left. \{ I_{pk}\cos(Df_k) - Q_{pk}\sin(Df_k) \} \right] \right\} \right) \\ & + \frac{1}{s_n^2 h_{eml}} \sum_{m=0}^{M} \left\{ \sum_{k=k_0+20m}^{k_0+20m+19} \frac{1}{N_k} \left[I_{emlk}^2 + Q_{emlk}^2 \right. \right. \\ & \qquad \left. \left. + \frac{N_k^2 A_0^2}{4} R_{eml}^2(Dt_k) \right] \right\} \\ & - \sum_{m=0}^{M} ln\left(2\cosh\left\{ \frac{1}{s_n^2 h_{eml}} \sum_{k=k_0+20m}^{k_0+20m+19} \left[A_0 R_{eml}(Dt_k) \right. \right. \right. \\ & \qquad \left. \left. \left. \{ I_{emlk}\cos(Df_k) - Q_{emlk}\sin(Df_k) \} \right] \right\} \right) \end{aligned} \quad (14)$$

This cost function is a negative log likelihood function. In other words, $C\exp[-J_a(x_{f0},x_{w0}, x_{a0},A_0,t_{s0})]$ is the joint probability density function for the four accumulation time histories I_{pk}, Q_{pk}, I_{emlk}, and Q_{emlk} for $k = k_0$ to $(k_0+20M+19)$ conditioned on the 5 unknown parameters x_{f0}, x_{w0}, x_{a0}, A_0, and t_{s0}, where C is a normalizing constant. The ln[2cosh()] terms are the results of probability summations over the uncertain data bits d_m under the

assumption that the +1 and –1 values are equally likely. The dependence of J_a on x_{f0}, x_{w0}, and x_{a0} comes partly through the carrier phase model in eq. (6), but with w_{fk} set to 0 because the batch filter does not consider process noise. This yields the model $Df_k(x_{f0},x_{w0},x_{a0}) = [x_{f0} + x_{w0}(t_{midk}-t_{NCOk0}) + 0.5x_{a0}(t_{midk}-t_{NCOk0})^2 - w_{NCO}t_{midk}]$. The dependence of J_a on t_{s0} comes through the Dt_k terms. They can be computed as functions of x_{w0}, x_{a0}, and t_{s0} by iterating eq. (9) with w_{fk} and w_{tsk} both set zero. This iteration generates the time series $t_{sk}(x_{w0},x_{a0},t_{s0})$ for $k = k_0$ to $(k_0+20M+19)$, and eq. (10) yields $Dt_k(x_{w0},x_{a0},t_{s0}) = [t_{sk+1}(x_{w0},x_{a0},t_{s0})+t_{sk}(x_{w0},x_{a0},t_{s0})]/2 - t_{midk}$.

An approximation to the cost function in eq. (14) is useful for estimating the carrier phase parameters x_{f0}, x_{w0}, and x_{a0} and the carrier amplitude A_0. It presumes that the code phase is correct, which implies that $Dt_k = 0$, $R_{eml}(Dt_k) = 0$, and $R(Dt_k) = 1$. This approximation makes the last two summations of J_a constant, and therefore irrelevant to any optimization. An additional approximation substitutes the absolute value function for the ln[2cosh()] function. Use of the absolute value function amounts to "hard" bit estimation, in which the bit sign is set equal to the sum over the data bit interval of $I_{pk}\cos(Df_k) - Q_{pk}\sin(Df_k)$. The cost function approximation is

$$J_b(x_{f0}, x_{w0}, x_{a0}, A_0) = \frac{1}{s_n^2}\sum_{m=0}^{M}\left\{\sum_{k=k_0+20m}^{k_0+20m+19}\frac{1}{N_k}\left[I_{pk}^2+Q_{pk}^2+\frac{N_k^2A_0^2}{4}\right]\right\} - A_0\frac{1}{s_n^2}\sum_{m=0}^{M}\left|\sum_{k=k_0+20m}^{k_0+20m+19}\left[I_{pk}\cos(Df_k)-Q_{pk}\sin(Df_k)\right]\right| + \text{constant} \quad (15)$$

The above cost function is quadratic in A_0. It can be minimized by first minimizing the following cost function

$$J_c(x_{f0}, x_{w0}, x_{a0}) = -\sum_{m=0}^{M}\left|\sum_{k=k_0+20m}^{k_0+20m+19}\left[I_{pk}\cos(Df_k)-Q_{pk}\sin(Df_k)\right]\right| \quad (16)$$

with respect to x_{f0}, x_{w0}, and x_{a0}. One then computes $A_0 = -2J_{cmin}/N_{tot}$ to optimize J_b in eq. (15), where $N_{tot} = N_{k0} + N_{k0+1} + \ldots + N_{k0+20M+19}$.

Yet another cost function needs to be defined in order to deal with the fact that the cost function in eq. (16) has many local minima. A search on a grid has to be performed in order to find the global minimum, and an alternate cost function allows a reduction from a 3-dimensional grid to a 2-dimensional grid. The alternate cost function is

$$J_d(x_{f0}, x_{w0}, x_{a0}) = \sum_{m=0}^{M}\frac{1}{N_m}\left\{\sum_{k=k_0+20m}^{k_0+20m+19}\left[I_{pk}\sin(Df_k)+Q_{pk}\cos(Df_k)\right]\right\}^2 \quad (17)$$

This cost function is a weighted sum of the norm squared of the projection of the first two elements of the measurement error vector n_{ym} of eq. (5) perpendicular to the signal direction. The importance of this cost function is that it can be minimized with respect to x_{f0} analytically, which reduces the dimensionality of the search grid over which one must seek the global minimum. In order to minimize J_d with respect to x_{f0}, define

$$I_{rm}(x_{w0}, x_{a0}) = \sum_{k=k_0+20m}^{k_0+20m+19}\left[I_{pk}\cos(Df_k-x_{f0})-Q_{pk}\sin(Df_k-x_{f0})\right] \quad (18a)$$

$$Q_{rm}(x_{w0}, x_{a0}) = \sum_{k=k_0+20m}^{k_0+20m+19}\left[I_{pk}\sin(Df_k-x_{f0})+Q_{pk}\cos(Df_k-x_{f0})\right] \quad (18b)$$

where the r subscript of the accumulations stands for "rotated". Neither of these accumulations depends on x_{f0} because Df_k-x_{f0} is independent of x_{f0}, and the value of x_{f0} that minimizes J_d for given values of x_{w0} and x_{a0} is

$$x_{f0opt}(x_{w0}, x_{a0}) = -\frac{1}{2}atan2\left[\left(2\sum_{m=0}^{M}\frac{I_{rm}Q_{rm}}{N_m}\right),\left(\sum_{m=0}^{M}\frac{[I_{rm}^2-Q_{rm}^2]}{N_m}\right)\right] \quad (19)$$

A Sequence of Fine Acquisition Optimizations. The first step in the fine acquisition algorithm is an approximate global minimization of $J_c(x_{f0},x_{w0},x_{a0})$. This procedure searches for the minimum on a grid in (x_{w0},x_{a0}) space and computes x_{f0} at each grid point by using eq. (19). Figure 1 explains why the phase x_{f0} that minimizes J_d will be a good approximation to the phase that minimizes J_c. The figure plots the rotated accumulations $I_p\sin(Df) + Q_p\cos(Df)$ on the vertical axis versus $I_p\cos(Df) - Q_p\sin(Df)$ on the horizontal axis for a case in which x_{w0} and x_{a0} are nearly correct. The minimization of J_c with respect to x_{f0} performs an additional rotation in order to maximize the spread of these points along the horizontal axis, which would tend to align the solid line with the horizontal axis. The minimization J_d, rotates x_{f0} in order to minimize the spread of these points along the vertical axis, which would tend to align the dash-dotted line with the vertical axis. Thus, these two minimizations tend to produce nearly the same optimal x_{f0} rotation, which is why the eq. (19) value of x_{f0} can be used to approximately optimize $J_c(x_{f0}, x_{w0}, x_{a0})$.

The extent and spacing of the search grid in (x_{w0},x_{a0}) space must be chosen carefully in order to get a good solution.

The x_{w0} grid should be centered at the coarse acquisition Doppler shift estimate, and it should extend in either direction as far as the possible uncertainty in the coarse Doppler shift. A good rule of thumb is to extend for ±25% of the pre-detection bandwidth of the coarse acquisition or ±100% of the Doppler grid spacing of the coarse acquisition, whichever is greater. For most of the examples of this paper a range of ±12.5 Hz has been used. The x_{a0} grid should be centered at zero. Recall that x_{a0} models acceleration. The extent of this grid should be chosen to reflect the range of possible LOS accelerations. For a geosynchronous receiver, the LOS acceleration comes mainly from gravitation and ranges up to 0.081 g, which translates into x_{a0} = 26.3 rad/sec^2 (4.2 Hz/sec). The actual search should expand to include a factor of safety which ensures that the global minimum does not fall outside of the grid. A range of ±33.6 rad/sec^2 (±5.3 Hz/sec) has been used in the present study.

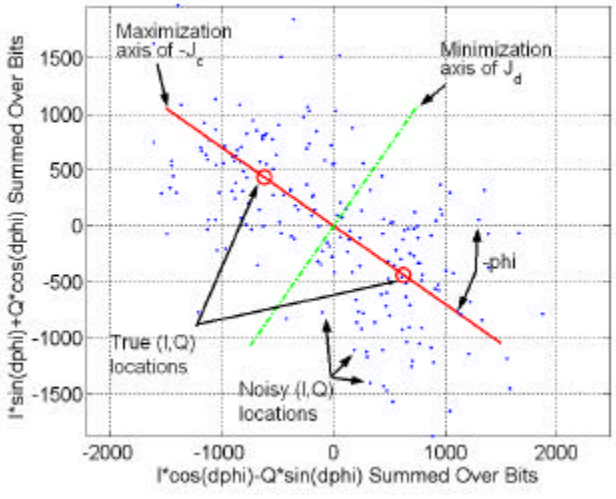

Fig. 1. Relationship of in-phase and quadrature accumulations to cost functions J_c and J_d, an 18 dB Hz example.

The required x_{w0} and x_{a0} grid spacings vary inversely with the length of the batch interval, T_{fine} = 0.020(M+1). The limits $Dx_{w0} \le p/(2T_{fine})$ and $Dx_{a0} \le p/(T_{fine}^2)$ ensure that the worst-case error between the true optimum and the closest (x_{w0},x_{a0}) grid value yields no more than a quarter cycle of erroneous I/Q rotation over the data batch. This bound on the erroneous rotation prevents the accumulations shown in Fig. 1 from being spun into a circular distribution that washes out the two distinct I/Q clouds whose detection is of central importance to phase estimation. For a typical batch duration of 3 seconds, these limits translate into a grid of 300 frequency points by 200 frequency-rate points, or 60,000 total points.

The second step of the fine acquisition algorithm computes the exact global minimum of $J_c(x_{f0},x_{w0},x_{a0})$. It does this using Newton's method. It starts with the 10 lowest isolated minima of $J_c[x_{f0opt}(x_{w0},x_{a0}), x_{w0}, x_{a0}]$ at the grid points of the previous calculation and performs an iterative Newton search from each point in order to optimize J_c with respect to x_{f0}, x_{w0}, and x_{a0}. The guarded Newton search includes precautions that ensure global convergence to a local minimum, such as modification of the cost Hessian and adaptation of the search step size [10]. The point that achieves the lowest minimum using Newton's method yields fine estimates for the carrier phase, x_{f0}, the Doppler shift, x_{w0}, and the Doppler shift rate, x_{a0}. Ten different initial guesses are used in order to increase the likelihood that one of the first guesses will reach the true global minimum of J_c.

Results from the first two steps of a typical fine acquisition are shown in Fig. 2. It depicts a contour of $J_b(x_{f0},x_{w0},x_{a0},A_0)$ at grid points in (x_{w0},x_{a0}) space with x_{f0} given by eq. (19) and A_0 given by the optimal formula that appears just after eq. (16). The cost $J_b(x_{f0},x_{w0},x_{a0},A_0)$, when optimized with respect to A_0, is monotonically related to $J_c(x_{f0},x_{w0},x_{a0})$; so, the J_b contour on Fig. 2 is also indicative of the behavior of J_c. The figure clearly shows many distinct local minima of the J_b contour. Their minimum costs decrease as they approach the true optimum. The final true optimum in the figure, shown as an asterisk, has been computed using Newton's method. It does not lie exactly at a grid point, and its cost is lower than that of the best grid point.

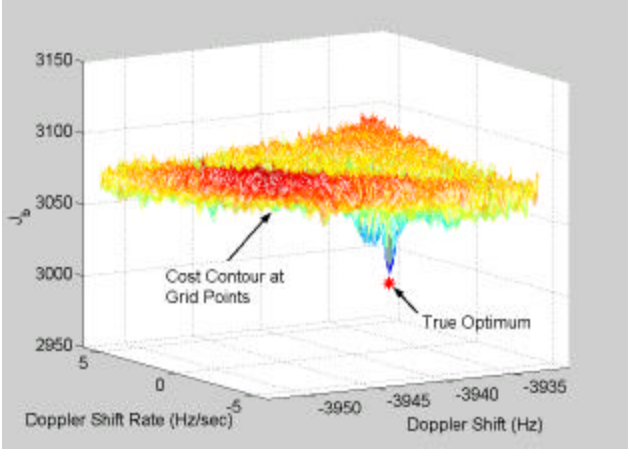

Fig. 2. Contour of J_b cost values and the true optimum as generated by the first 2 steps of the fine acquisition algorithm.

The third step of the fine acquisition makes additional improvements to the estimates of x_{f0}, x_{w0}, x_{a0}, A_0, and t_{s0} and computes an approximation of the Cramer-Rao lower bound on the estimation error covariance. The final estimates are determined by performing a guarded Newton search for the minimum of $J_a(x_{f0},x_{w0},x_{a0},A_0,t_{s0})$ from eq. (14). The first guesses are the outputs of the second step of the fine acquisition, except that the t_{s0}

guess is taken from the coarse acquisition. This search converges rapidly, normally in 3-10 iterations. The approximate Cramer-Rao estimation error covariance is the inverse of the Hessian of the J_a cost function at the final optimum, $P_{xx0} = [\partial^2 J_a/\partial x^2]^{-1}$, where x is the 5x1 vector $[x_{f0},x_{w0},x_{a0},A_0,t_{s0}]^T$.

Mid-Point Calculations for Kalman Filter Initialization. The fine acquisition algorithm's carrier phase and Doppler shift estimates have their highest accuracy at the mid-point of the acquisition batch interval. Therefore, the estimates at the mid point are used to initialize the tracking Kalman filter. The value m_c = round(M/2) is the index of the navigation bit whose start time is nearest to the midpoint of the fine acquisition batch interval. The start time of this bit is $t_{NCOk0+20mc}$, and $Dt_c = t_{NCOk0+20mc} - t_{NCOk0}$ is the time offset of the midpoint from the start of the fine acquisition batch interval. The Kalman filter initializes at time $t_{k0+20mc}$ using the initial state estimate $[(x_{f0}+Dt_c x_{w0}+0.5Dt_c^2 x_{a0}),(x_{w0}+Dt_c x_{a0}), x_{a0},A_0,t_{sc}]^T$, where t_{sc} is the estimated start time of the mid-point navigation data bit as determined by iteration of eq. (9) with zero process noise. The covariance gets propagated to this time point using the following formula:

$$P_{xxc} = TP_{xx0}T^T, \text{ where } T = \begin{bmatrix} 1 & Dt_C & Dt_C^2/2 & 0 & 0 \\ 0 & 1 & Dt_C & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

This propagation neglects the effects of x_{w0} and x_{a0} uncertainty on t_{sc}, which is reasonable because of their small magnitude.

IV. AN EXTENDED KALMAN FILTER FOR CARRIER AND CODE TRACKING

The extended Kalman filter tracking algorithm is a straight-forward implementation of Kalman filtering principles, except for two points. First, it uses a Bayesian integration process to deal with the uncertain data bits. Second, it uses nonlinear iteration in a somewhat unconventional way during its measurement update. The first part of this section describes how the Kalman filter operates when the navigation data bit sign is assumed to be known. The second part explains how a Bayesian analysis deals with data bit uncertainty by mixing the different estimates that result from different assumptions about data bit signs.

Iterated Extended Kalman Filter with Assumed Bit Signs. The iterated extended Kalman filter performs a single measurement update and state propagation over a single data bit interval by solving the following weighted least squares problem:

find: x_m, x_{m+1}, and w_m (21a)

to minimize:

$$J = \frac{1}{2}[\tilde{R}_{xxm}(x_m - \tilde{x}_m)]^T[\tilde{R}_{xxm}(x_m - \tilde{x}_m)] + \frac{1}{2}(R_{wwm}w_m)^T(R_{wwm}w_m) + \frac{1}{2}[y_m - d_m h_m(x_m, w_m)]^T[y_m - d_m h_m(x_m, w_m)] \quad (21b)$$

subject to:

$$x_{m+1} = f_m(x_m,w_m) \quad (21c)$$

where the unknown solution vectors are the state $x_m = [x_{fm},x_{wm},x_{am},A_m,t_{sm}]^T$ and the process noise $w_m = [w_{fm}^T,w_{tsm},w_{Am}]^T$. The known data bit value is d_m. The measurement function $h_m(x_m,w_m)$ is effectively defined by eq. (5) with eqs. (7), (10), and (12) used to substitute for Df_m, Dt_m, and $\bar{A}_m$ in terms of x_m and w_m. The discrete-time dynamics function $f_m(x_m,w_m)$ is effectively defined by eqs. (6), (9), and (11). It is linear except for the carrier aiding term in the t_{sm} iteration, eq. (9), and the nonlinear term in that equation can be well approximated by a linear model because w_{L1} is much larger than the Doppler shift.

The vector $\tilde{x}_m$ is the a priori estimate of x_m based on all of the accumulations up through data bit interval m-1, the matrix $\tilde{R}_{xxm}$ is the corresponding a priori estimation error square root information matrix, and the matrix R_{wwm} is the a priori process noise square root information matrix. The corresponding a priori covariances are related to these matrices as follows: $P_{xxm} = \tilde{R}_{xxm}^{-1}\tilde{R}_{xxm}^{-T}$ and $P_{wwm} = R_{wwm}^{-1}R_{wwm}^{-T}$, where the notation $()^{-T}$ refers to the inverse transpose of the matrix in question. Note that $\tilde{x}_m$ and $\tilde{R}_{xxm}$ correspond to the batch fine acquisition algorithm's mid-interval estimate for m = 0, and they are determined by the previous iteration of the Kalman filter for $m > 0$. R_{wwm} is determined from P_{wwm}, which is defined by the various process noise covariances already given in eq. (8) and in the text sections that follow eqs. (9) and (11) and by the fact that w_{fm}, w_{tsm}, and w_{Am} are uncorrelated.

The Kalman filter solution procedure first minimizes the cost function in eq. (21b) with respect to x_m and w_m, and then it uses eq. (21c) to propagate the solution to x_{m+1}. The minimization in eq. (21b) is an iterative Newton minimization that is guarded in order to ensure convergence. This differs from a conventional iterative Kalman filter because it uses step-size adaptation to ensure convergence to a local minimum of a weighted least-squares cost function.

The Hessian matrix of the cost function in eq. (21b), which is required for Newton's method and for calculation of $\tilde{R}_{xxm+1}$, is computed and stored in square

root form. This procedure first computes the upper-triangular square root of the Gauss-Newton approximation of the Hessian, R_G. This calculation is performed using QR factorization:

$$Q_G \begin{bmatrix} R_G \\ 0 \end{bmatrix} = \begin{bmatrix} R_{wwm} & 0 \\ 0 & \tilde{R}_{xxm} \\ (d_m \frac{\partial h_m}{\partial w}) & (d_m \frac{\partial h_m}{\partial x}) \end{bmatrix} \quad (22)$$

Next, the procedure calculates the exact Hessian in the factored form $R_G^T D R_G$, where the symmetric D matrix differs from the identity by a term that involves the second derivatives of h_m with respect to x_m and w_m. The procedure next attempts to Cholesky factorize D into the form $R_D^T R_D = D$. If this factorization fails because D is not positive definite, then the Hessian square root R_H gets approximated as $R_H = R_G$. Otherwise, the Hessian square root is computed exactly as $R_H = R_D R_G$. Standard Newton techniques then use R_H and the gradient of the cost function to compute updates to x_m and w_m.

The last operation of the Kalman filter is the propagation to determine $\tilde{x}_{m+1}$ and $\tilde{R}_{xxm+1}$ so that the algorithm can operate recursively starting at the next data bit interval. The solution to the optimization problem in eq. (21b), the cost Hessian evaluated at the solution, and eq. (21c) are used to do the propagation. Suppose that the estimates $\hat{x}_m$ and $\hat{w}_m$ minimize the cost in eq. (21b) and suppose that the square root of the Hessian is

$$\begin{bmatrix} \hat{R}_{wwm} & \hat{R}_{wxm} \\ 0 & \hat{R}_{xxm} \end{bmatrix} = R_H \quad (23)$$

Suppose, also, that the Jacobian of the dynamics function in eq. (21c) is

$$\begin{bmatrix} G_m & F_m \end{bmatrix} = \left[\begin{matrix} \frac{\partial f_m}{\partial w} & \frac{\partial f_m}{\partial x} \end{matrix} \right]_{(\hat{x}_m, \hat{w}_m)} \quad (24)$$

Then $\tilde{x}_{m+1}$ is determined by evaluation of eq. (21c) at the current best estimate, i.e., $\tilde{x}_{m+1} = f_m(\hat{x}_m, \hat{w}_m)$, and $\tilde{R}_{xxm+1}$ is computed via the following QR factorization [11,12]:

$$Q_m \begin{bmatrix} \tilde{R}_{wwm} & \tilde{R}_{wxm} \\ 0 & \tilde{R}_{xxm+1} \end{bmatrix} = \begin{bmatrix} (\hat{R}_{wwm} - \hat{R}_{wxm} F_m^{-1} G_m) & \hat{R}_{wxm} F_m^{-1} \\ (-\hat{R}_{xxm} F_m^{-1} G_m) & \hat{R}_{xxm} F_m^{-1} \end{bmatrix} \quad (25)$$

where Q_m is an orthonormal matrix and $\tilde{R}_{wwm}$ and $\tilde{R}_{xxm+1}$ are square, non-singular, upper-triangular matrices.

The square-root Kalman filter form used in these developments is important to successful implementation of this tracking algorithm. In theory, it would be possible to develop a non-square root filter, but experience with this particular problem has shown that non-square root filters can diverge due to the build-up of numerical round-off errors. Square root filters are specifically tailored to alleviate such problems [11].

Bayesian Treatment of Unknown Navigation Data Bits. The Kalman filter of the preceding sub-section, which assumes that the data bit sign is known, can be used as part of a Bayesian analysis when the bit sign is unknown. The Bayesian approach executes the filter calculations twice, once for each possible navigation data bit sign. Suppose that the results are $\tilde{x}_{m+1}^{(+)}$ and $\tilde{R}_{xxm+1}^{(+)}$ for the assumption that $d_m = +1$ and $\tilde{x}_{m+1}^{(-)}$ and $\tilde{R}_{xxm+1}^{(-)}$ for $d_m = -1$. Suppose, also, that the eq. (21b) optimal costs for these two different solutions are, respectively, $J_m^{(+)}$ and $J_m^{(-)}$ and suppose that the a priori probabilities of +1 and –1 bit values are $p_m^{(+)} = p_m^{(-)} = 0.5$. Then it can be shown that the a posterior probabilities of the +1 and –1 bit values are

$$\tilde{p}_{m+1}^{(+)} = \frac{r_m^{(+)}}{r_m^{(+)} + r_m^{(-)}}, \quad \tilde{p}_{m+1}^{(-)} = \frac{r_m^{(-)}}{r_m^{(+)} + r_m^{(-)}} \quad (26)$$

where

$$r_m^{(+)} = \frac{p_m^{(+)} \left| det(\tilde{R}_{xxm}) \right|}{\left| det(\tilde{R}_{xxm+1}^{(+)}) \right|} exp\{min[J_m^{(+)}, J_m^{(-)}] - J_m^{(+)}\} \quad (27a)$$

$$r_m^{(-)} = \frac{p_m^{(-)} \left| det(\tilde{R}_{xxm}) \right|}{\left| det(\tilde{R}_{xxm+1}^{(-)}) \right|} exp\{min[J_m^{(+)}, J_m^{(-)}] - J_m^{(-)}\} \quad (27b)$$

The derivation of these formulas uses the conditional probability density function for x_m given the measurements and the bit sign assumption. This probability density equals $C\exp(-J)$, where C is a normalizing constant and J is the cost from eq. (21b). The derivation approximates J by a quadratic expansion about its minimum. The leading term in the exponential argument in eqs. (27a) and (27b), $min[J_m^{(+)}, J_m^{(-)}]$, is not needed in theory, but it is useful in practice as a means of avoiding computer overflow or underflow problems.

One can see from eqs. (26)-(27b) that the optimal cost is the important factor in determining the a posteriori probabilities of the two bit signs. All other things being equal, the bit sign assumption that produces the smallest optimal cost in eq. (21b) will have the highest a posteriori probability. This makes intuitive sense because it gives preference to the bit sign whose corresponding a priori expected measurement is closest

to the actual measurement.

The probabilities in eq. (26) can be used to "mix" the two estimates and their square-root information matrices to produce the expected state and the covariance of the estimation error in the expected state. This mixing presumes that the estimation error distributions for the two bit sign assumptions are Gaussian. The mixed values are:

$$\tilde{x}_{m+1} = \tilde{p}_{m+1}^{(+)}\,\tilde{x}_{m+1}^{(+)} + \tilde{p}_{m+1}^{(-)}\,\tilde{x}_{m+1}^{(-)} \tag{28a}$$

$$\tilde{P}_{m+1} = \tilde{p}_{m+1}^{(+)}[\,\tilde{P}_{m+1}^{(+)} + (\,\tilde{x}_{m+1}^{(+)} - \tilde{x}_{m+1})(\,\tilde{x}_{m+1}^{(+)} - \tilde{x}_{m+1})^{\mathrm{T}}] + \tilde{p}_{m+1}^{(-)}[\,\tilde{P}_{m+1}^{(-)} + (\,\tilde{x}_{m+1}^{(-)} - \tilde{x}_{m+1})(\,\tilde{x}_{m+1}^{(-)} - \tilde{x}_{m+1})^{\mathrm{T}}] \tag{28b}$$

Thus, the mixed state estimate is a simple weighted sum of the two state estimates. The mixed covariance estimate includes a weighted sum of the two covariances, but it also includes terms that increase the covariance in the directions from the mixed estimate to the estimates that apply if one or the other bit sign assumption is correct. This makes sense because one or the other of these estimates would be the best one if the bit sign were known, which increases the uncertainty in these directions. Of course, if one of the a posteriori probabilities is very near 1, then the mixed state estimate and covariance are very nearly equal to the state estimate and the covariance for that bit sign assumption

A square-root information version of eq. (28b) has been developed. The square root mixing method uses QR factorization to calculate the square, non-singular, upper-triangular matrix R_{mix} and the orthonormal matrix Q_{mix} that satisfy the following relationship:

$$[R_{mix} \quad 0]Q_{mix} = \Big[I, \quad \tilde{R}_{xxm+1}^{(+)}\{\tilde{x}_{m+1}^{(+)} - \tilde{x}_{m+1}\}, \ldots \sqrt{\frac{\tilde{p}_{m+1}^{(-)}}{\tilde{p}_{m+1}^{(+)}}}\,\tilde{R}_{xxm+1}^{(+)}\{\tilde{R}_{xxm+1}^{(-)}\}^{-1}, \ldots \sqrt{\frac{\tilde{p}_{m+1}^{(-)}}{\tilde{p}_{m+1}^{(+)}}}\,\tilde{R}_{xxm+1}^{(+)}\{\tilde{x}_{m+1}^{(-)} - \tilde{x}_{m+1}\}\Big] \tag{29}$$

The following formula then gives the mixed square-root information matrix:

$$\tilde{R}_{xxm+1} = \sqrt{\frac{1}{\tilde{p}_{m+1}^{(+)}}}\,R_{mix}^{-1}\tilde{R}_{xxm+1}^{(+)} \tag{30}$$

The preceding calculations presume that $\tilde{p}_{m+1}^{(+)} \geq \tilde{p}_{m+1}^{(-)}$. If $\tilde{p}_{m+1}^{(+)} < \tilde{p}_{m+1}^{(-)}$, then eqs. (29) and (30) get modified by interchanging the (+) and (-) superscripts.

These Bayesian calculations are critical to the successful operation of the tracking Kalman filter at low carrier-to-noise densities. It can become difficult to determine bit signs exactly when the signal is weak. Consider Fig. 1. Bit signs in this example are estimated according to which side of the dash-dotted line each (*I*,*Q*) point lies on. One can see that the signs of a number of bits will be mis-identified because the two clouds of (*I*,*Q*) points intersect. This intersection is the defining characteristic of the low carrier-to-noise density case. If there is a carrier phase error, then (*I*,*Q*) points near the separating line tend to get mis-identified in a biased way that can destabilize a PLL. The Bayesian method de-weights such (*I*,*Q*) combinations because their $\tilde{p}_{m+1}^{(+)}$ and $\tilde{p}_{m+1}^{(-)}$ values tend to be nearly equal. This de-weighting helps to counteract the destabilizing effect of systematic bit mis-identification.

There are other possible algorithms for dealing with the uncertain navigation data bits. A number of them were tried before the Bayesian method was developed. One method chose d_m based on which value gave the lowest eq. (21b) cost at $x_m = \tilde{x}_m$ and $w_m = 0$. Another method made a "soft" bit sign decision by including an ln[2cosh()] term in the measurement error part of eq. (21b). Neither of these methods performed as well as the Bayesian method. They all suffered from a propensity to make too definite of a decision about the correct bit sign when the (*I*,*Q*) point was near the sign boundary as determined from the imperfect carrier phase estimate.

Explicit Multi-Bit Bayesian Analysis. It is possible to extend the analysis of the previous sub-section to deal with multiple data bits in a Bayesian manner. Suppose that N_b is the total number of data bits whose signs are explicitly allowed to vary in the Bayesian analysis. Both possibilities for the current navigation data bit sign are considered along with both possibilities for the most recent N_b-1 navigation data bits. Each of the 2^{Nb-1} different possible combinations of these recent bits gives rise to an a priori state estimate and an estimation error information matrix, $\tilde{x}_m^{(i)}$ and $\tilde{R}_{xxm}^{(i)}$ for i = 1,…, 2^{Nb-1}, and each of these estimates has an a priori probability associated with it, $\tilde{p}_m^{(i)}$.

The estimation procedure computes 2^{Nb} Kalman filter estimates and square-root information matrices along with their probabilities, $\tilde{x}_{m+1}^{(i,+)}$, $\tilde{R}_{xxm+1}^{(i,+)}$ and $\tilde{p}_{m+1}^{(i,+)}$ for i = 1,…, 2^{Nb-1}, and $\tilde{x}_{m+1}^{(i,-)}$, $\tilde{R}_{xxm+1}^{(i,-)}$ and $\tilde{p}_{m+1}^{(i,-)}$ for i = 1,…, 2^{Nb-1}. In this formulation $\tilde{x}_{m+1}^{(i,j)}$, $\tilde{R}_{xxm+1}^{(i,j)}$ and $\tilde{p}_{m+1}^{(i,j)}$ are the Kalman filter state, the square-root information matrix, and the associated a posteriori probability under the assumptions that the correct a priori state variable estimate is represented by $[\,\tilde{x}_m^{(i)}, \tilde{R}_{xxm}^{(i)}\,]$ and that the correct current bit sign is $d_m = j$. The a posteriori probabilities of the different cases are calculated using

modified versions of eqs. (26)-(27b): The sum in the denominator of eq. (26) is taken over all 2^{Nb} r values, the formulas for r in eqs. (27a) and (27b) get multiplied by $\tilde{p}_m^{(i)}$, the term $|det(\tilde{R}_{xxm})|$ in eqs. (27a) and (27b) changes to $|det(\tilde{R}_{xxm}^{(i)})|$, and the minimal cost in the exponentials in eqs. (27a) and (27b) is taken over all 2^{Nb} cases. Equations (28a) and (28b) also change in the multi-bit formulation. They compute weighted sums over all 2^{Nb} cases, and there is a corresponding change to the square-root information matrix update in eqs. (29) and (30).

This approach requires an additional mixing operation in order to eliminate explicit consideration of the oldest data bit. Suppose that $\tilde{x}_m^{(i)}$, $\tilde{R}_{xxm}^{(i)}$, and $\tilde{p}_m^{(i)}$ correspond to one particular set of assumptions about the previous N_b-1 data bit signs, and that $\tilde{x}_m^{(i+1)}$, $\tilde{R}_{xxm}^{(i+1)}$, and $\tilde{p}_m^{(i+1)}$ corresponds to identical assumptions except that the sign of the oldest data bit is reversed. Then the pair of solutions $[\tilde{x}_{m+1}^{(i,+)}, \tilde{R}_{xxm+1}^{(i,+)}, \tilde{p}_{m+1}^{(i,+)}]$ and $[\tilde{x}_{m+1}^{(i+1,+)}, \tilde{R}_{xxm+1}^{(i+1,+)}, \tilde{p}_{m+1}^{(i+1,+)}]$ needs to get combined in order to keep the number of solutions from growing. This must be done for all pairs whose only difference of bit sign assumption is in the oldest bit. Each such pair gets mixed using eqs. (28a) and (29), except the mixing probabilities used in those equations are $\tilde{p}_{m+1}^{(i,+)}/[\tilde{p}_{m+1}^{(i,+)} + \tilde{p}_{m+1}^{(i+1,+)}]$ and $\tilde{p}_{m+1}^{(i+1,+)}/[\tilde{p}_{m+1}^{(i,+)} + \tilde{p}_{m+1}^{(i+1,+)}]$. The final probability that gets assigned to the mixed state estimate and the mixed square-root information matrix is $\tilde{p}_{m+1}^{(i,+)} + \tilde{p}_{m+1}^{(i+1,+)}$.

The advantage of the multi-bit method is that it makes fewer approximations. Use of the mixed state vector estimate from eq. (28a) and the mixed square root information matrix from eq. (29) in the next iteration of the filter represents an approximation. This approximation does not account for uncertainty about past bit signs as accurately as does the multi-bit method. Thus, there is hope that the multi-bit method may be better able to avoid carrier cycle slips and loss of PLL lock. The possibility of increased performance comes at a cost. The amount of computation grows as 2^{Nb}, which is the number of parallel Kalman filters that must be run simultaneously. Therefore, it is unlikely that $N_b > 5$ or 6 could be used practically, and it would be preferable if the original 1-bit algorithm would suffice.

PLL and DLL Feedback Control Laws. Ad hoc feedback control laws are needed in order to complete the DLL and PLL designs [8]. These feed the Kalman filter's state estimates back to drive the carrier and code NCOs. Suitable feedback control laws take the form:

$$t_{NCOm+3} = \tilde{t}_{sm+1} + \frac{w_{L1} dt_{nom}}{w_{L1} + \tilde{x}_{wm+1} + 0.5 dt_{m+1} \tilde{x}_{am+1}} + \frac{w_{L1} dt_{nom}}{w_{L1} + \tilde{x}_{wm+1} + 1.5 dt_{m+1} \tilde{x}_{am+1}} \quad (31a)$$

$$w_{NCOm+2} = \{(1-g^2)\tilde{x}_{fm+1} + [dt_{m+1}(1-2g) + dt_{m+2}]\tilde{x}_{wm+1} + 0.5[(dt_{m+1}+dt_{m+2})^2 - 2g dt_{m+1}^2]\tilde{x}_{am+1} - dt_{m+1}(1-2g) w_{NCOm+1}\}/dt_{m+2} \quad (31b)$$

where eq. (31a) is the DLL feedback and eq. (31b) is the PLL. The parameter g is a PLL tuning parameter that must be in the range $0 \le g < 1$. Values near 1 give a slow response and are preferred in the weak signal case. The performance of the Kalman filter is insensitive to this tuning parameter so long as it is neither too small, which could produce jerky response and possible aliasing problems, nor too near 1, which could allow the NCO Doppler errors to become large enough to adversely affect signal strength. g = 0.9798 has been used successfully in many of the cases that are discussed in Section V.

These algorithms are completely causal. They can be implemented on-line given sufficient processor power. The accumulations up through navigation bit interval m are used to estimate the carrier phase, carrier amplitude, and code phase states at time t_{NCOm+1}, which is the end point of the bit interval. These estimates, in turn, are used to derive NCO frequencies for the DLL and the PLL for the navigation data bit interval that starts at time t_{NCOm+2}. The Kalman filter computations and the feedback calculations can be performed during the data bit interval that starts at time t_{NCOm+1} and that ends at time t_{NCOm+2} because the necessary inputs are available at the start of this interval, and the outputs are not needed until after it ends.

V. SIMULATION TESTS OF FINE ACQUISITION AND KALMAN FILTER TRACKING ALGORITHMS

Off-Line Signal Simulation. The fine acquisition and Kalman filter tracking algorithms have been tested using a high-fidelity simulation of the outputs from an RF front-end. The simulated data have been generated off-line in MATLAB along with truth values of the signal parameters of interest. The simulation includes the effects of receiver thermal noise, receiver clock drift, 2-bit digitization with automatic gain control, delay and code distortion caused by the RF front-end's band-pass filter, random-walk LOS acceleration, random-walk code-carrier divergence, random-walk carrier amplitude variations, and interfering GPS signals. The reported C/N_0 values for these simulations are computed at the input to the digitizer, and 1 dB is subtracted from the result in order to account for loss during optimal 2-bit digitization [2].

Two receiver clock models have been used in these simulations. One models a temperature-compensated crystal oscillator (TXCO) and has worse performance than the other. Referring to eq. (8), its characteristics are defined by the parameters $S_f = 5\times10^{-21}$ sec and $S_g = 5.9\times10^{-20}$/sec, which gives it a minimum root Allan variance of 1.4×10^{-10} at $t_{min} = 0.5$ sec [9]. The other clock model represents an ovenized crystal oscillator (OXO) that is known to have flown in a space-qualified receiver. Its characteristics are $S_f = 5\times10^{-23}$ sec and $S_g = 1.5\times10^{-22}$/sec, which yield a minimum root Allan variance = 1.0×10^{-11} at $t_{min} = 1.0$ sec.

Two different values of the LOS acceleration random walk intensity have been used. One model uses $q_{LOS} = 0.005$ rad^2/sec^5. This is somewhat representative of the LOS acceleration variations that occur at GEO due to orbital motion of the GPS satellites. It exaggerates the effects of LOS dynamics for time scales shorter than 1,200 sec, and it under-estimates their effects over longer time scales. The important time scale to consider is the response time of the Kalman filter tracker, which is on the order of 1 sec. Thus, 0.005 rad^2/sec^5 is a conservative value for q_{LOS}. Another value that has been used is $q_{LOS} = 0.5$ rad^2/sec^5, which is very conservative. It provides an extreme test condition for the tracking Kalman filter.

Fine Acquisition Results. Consider the typical fine acquisition case whose carrier phase error time history is shown in Fig. 3. This case corresponds to $C/N_0 = 18$ dB Hz. It uses the poorer receiver clock and the larger of the two q_{LOS} acceleration random walk intensities. The batch interval is 3 seconds long. The horizontal time axis of the plot is measured from the center of the fine acquisition data batch and covers the entire batch interval. The two dash-dotted sigma curves plot the ± 1 *s* estimation error limits as determined by the batch filter.

Figure 3 is representative of all the fine acquisition results. The solid carrier phase error curve is roughly cubic. Its peak errors occur near the end points of the batch interval, and its magnitude is relatively small near the center of the interval. The mid-point error is 14°; so, it provides a very good phase initialization for the Kalman filter tracking algorithm. The batch filter's predicted *s* underestimates the actual error standard deviation, which makes sense when one recalls that the batch filter does not include a process noise model. The corresponding carrier frequency error, not shown, has a maximum value of 0.6 Hz near the beginning of the batch interval, but it is only 0.2 Hz at the mid point, where the Kalman filter initializes. The mid-point code-phase and carrier-amplitude errors are, respectively, 0.022 chips and 0.6 dB. This fine acquisition's mid-point estimates provide an excellent initialization for the Kalman filter tracking algorithm.

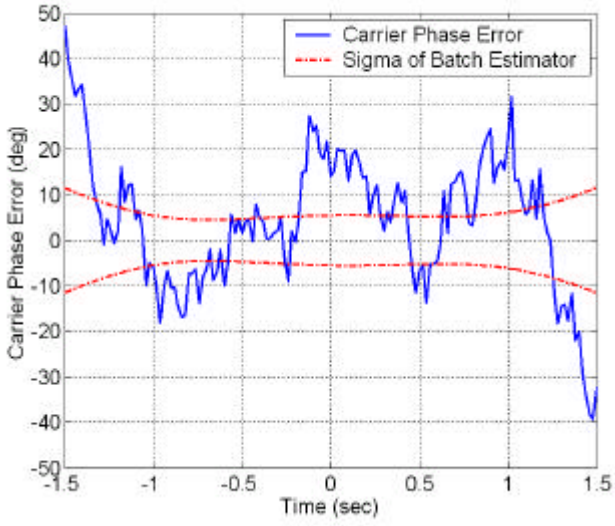

Fig. 3. Carrier phase error time history for a fine acquisition plotted over the acquisition interval, TCXO receiver clock & 18 dB Hz C/N_0.

Batch fine acquisitions have been tested for C/N_0 values ranging from 15 to 24 dB Hz. A batch interval of 6 seconds has been used for the 15 dB Hz cases, and intervals of 3 seconds have been used for most of the other cases. Of 14 cases that have been tried, only 3 showed any anomalies, and only 1 showed a serious anomaly. Each anomaly took the form of a single half-cycle slip of the carrier phase error during the fine acquisition batch interval. In two of these cases, the slips occurred near an end of the fine acquisition interval, and the mid-point estimates were accurate for all quantities of interest. Thus, these cycle slips posed no apparent risk to the integrity of the acquisition. These 2 cases occurred at C/N_0 values of 20 and 22 dB Hz and used the poorer TCXO receiver clock. The one problematic case occurred at $C/N_0 = 15$ dB Hz when using the TCXO. The half-cycle slip occurred near the mid-point of the batch acquisition interval in this case, which made the calculation unreliable.

The fine acquisition results can be summarized as follows. This summary excludes the one anomalous case with a half-cycle slip near its mid-point. The maximum mid-point errors were 15° for carrier phase, 0.2 Hz for carrier Doppler shift, 0.053 chips for code phase, and 0.8 dB for carrier amplitude. The carrier phase, carrier frequency, and code phase errors tend to be smaller when the better clock, the OXO, is used. For long batch intervals, on the order of 6 seconds, a lower LOS acceleration random walk intensity also improves the accuracy of the phase and frequency estimation results. The most important result of all is that each fine

acquisition successfully initialized the Kalman filter tracker, as demonstrated by the fact that the tracker maintained lock for a significant interval following the initialization.

Kalman Filter Tracking Results. The Kalman filter's performance is illustrated by the results of 2 typical cases. Figure 4 presents the carrier phase error for these two cases. Both cases use the poorer receiver clock, the TCXO, the high-dynamics assumption for the LOS acceleration random-walk intensity, and C/N_0 = 18 dB Hz. In fact, both cases track exactly the same RF front-end data. The only difference is that the top plot explicitly considers data bit sign uncertainty only for the current bit (N_b = 1), whereas the bottom plot explicitly considers the 6 most recent data bit signs (N_b = 6). The N_b = 1 case maintains Doppler shift and code lock, but it experiences numerous carrier cycle slips, 1 two-cycle slip, 2 one-cycle slips, and 3 half-cycle slips, all in just 150 seconds of operation. The N_b = 6 case, on the other hand, experiences just 1 half-cycle slip in 150 seconds with a few near slips as well. The Doppler shift errors show similar benefits from an increase in N_b. The Doppler shift errors are 2.05 Hz max and 0.33 Hz RMS for the N_b = 1 case, but they diminish to 1.11 Hz max and 0.27 Hz RMS when N_b = 6. Thus, the explicit consideration of more than 1 data bit sign makes the tracker more robust when operating on weak signals.

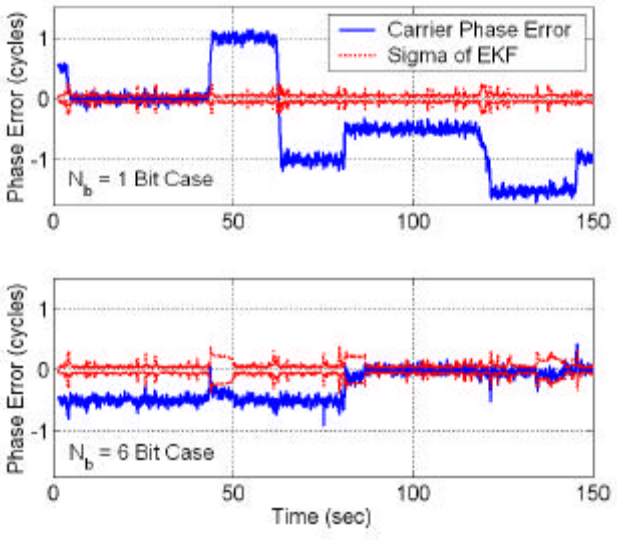

Fig. 4. EKF carrier phase tracking error time histories for two cases, TCXO receiver clock & 18 dB Hz C/N_0.

The Kalman filter's computed estimation error standard deviation indicates when there are possible cycle slip problems. The filter's standard deviation is derived by using the $\tilde{R}_{xxm}$ square-root information matrix to compute the estimation error covariance matrix. The dotted curves on Fig. 4 are plus and minus plots of the filter's computed carrier phase estimation error standard deviation. These curves show spikes of about ¼ cycle at the time of every cycle slip except for the last slip on the top plot. On the bottom plot, the standard deviation curves show periods with elevated uncertainty whenever the actual estimation error is experiencing some sort of anomaly. Thus, the filter's statistical model of its performance is reasonable and informative, especially when N_b is greater than 1.

The EKF tracking algorithm has been tried on a number of cases. If the filter uses N_b = 1 bit and if the poorer TCXO receiver clock is used, then the EKF maintains full lock down to C/N_0 = 22 dB Hz. Below this threshold cycle slips start to crop up. The EKF maintains code lock and Doppler lock,. but with cycle slips, down to C/N_0 = 18 dB Hz. At 15 dB Hz it eventually suffers a total loss of lock. If N_b increases to 4, then the tracker maintains total lock down to C/N_0 = 20 dB Hz. The frequency and average size of the cycle slips get reduced at 18 dB Hz when N_b = 4, but the EKF still suffers a total loss of lock at 15 dB Hz. These results are summarized in Fig. 5, which plots the tracker's RMS carrier phase error vs. the carrier-to-noise density, C/N_0, when using N_b = 1 and the TCXO receiver clock. The dash-dotted curve with the circle markers gives the filter's theoretical carrier phase error standard deviation, the dotted curve with the square markers gives the actual RMS carrier phase errors, and the solid curve with the triangle markers gives the actual RMS carrier phase errors after the cycle slips have been artificially removed. This figure shows that the system maintains full lock down to 22 dB Hz and that its theoretical performance matches its actual performance even down to 18 dB Hz once cycle slips have been accounted for. An additional result is plotted at C/N_0 = 20 dB Hz for the case of N_b = 4. These points show that the filter maintains lock in this case and that the actual and theoretical performance closely match each other. The N_b = 1 tracker maintains lock to a threshold that is 3 dB Hz below the tracking threshold for the PLL considered in Ref. 2, and the lock threshold of the N_b = 4 tracker is 5 dB Hz below that of Ref. 2's PLL.

If the random dynamic variations of the carrier phase get reduced, then the situation improves markedly. If the better receiver clock, the OXO, gets used and if the LOS acceleration random walk intensity is reduced to a moderately conservative value for GEO, then the N_b = 1 tracking algorithm maintains full lock with no carrier cycle slips down to C/N_0 = 15 dB Hz and possibly even below that. This represents a significant performance improvement over the 23 dB Hz threshold for loss of PLL lock given in Ref. 2 for the case of a perfect receiver clock.

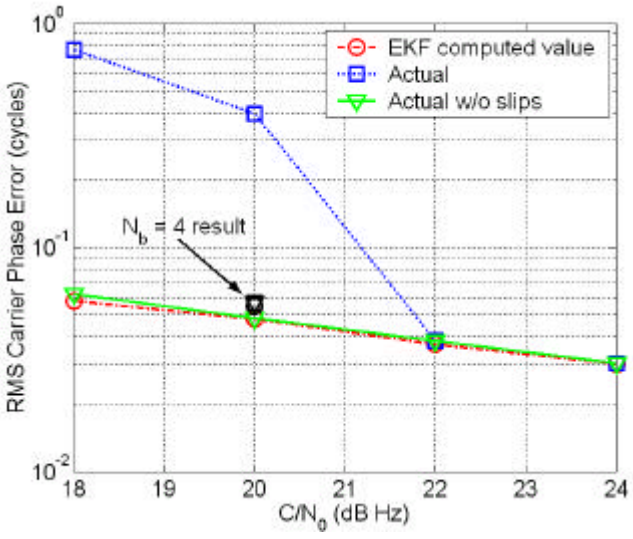

Fig. 5. EKF carrier phase tracking error performance as a function of C/N_0, using N_b = 1 and the TCXO.

Bit Errors, Sub-Frame Lock, and Autonomous Bit Aiding. When there are no cycle slips, the bit error rates for these new tracking algorithms approach the theoretical bit error rates that would exist with perfect carrier tracking. For example, when C/N_0 = 15 dB Hz a perfect tracker would experience a bit error rate of 13.0%. The new Kalman filter has a bit error rate of 14.1% when tracking a 15 dB Hz signal that has the benign phase dynamics of an OXO receiver clock and geosynchronous LOS accelerations.

Bit recognition becomes more problematic when cycle slips occur, but the decoded bits can be used to achieve sub-frame lock even during times of significant cycle slips. Suppose that one correlates the known preamble, time-of-week (TOW), and 2nd-word parity bits of each sub-frame with the tracker's decoded bits. Suppose, also, that one adds up these correlations over a number of sub-frames, but only after taking the absolute value of each sub-frame's correlation in order to undo the effects of possible half-cycle slips. Then the resulting correlation function will have a peak when the known true bits are aligned correctly with the decoded bits. Figure 6 shows the results of such a calculation. It corresponds to the same case as the top plot on Fig. 4, the case with C/N_0 = 18 dB Hz, N_b = 1, and a TCXO. The absolute values of the correlations have been summed over 23 sub-frames. This correlation has an obvious peak at the correct correlation time, which means that sub-frame lock is achieved.

The ephemeris data bits can be determined in a straightforward manner when there are no cycle slips. This is done by voting the different values of the same bit that get collected from different 30-second frames. Suppose that the bit error rate probability is p_{err} and suppose that the bit sign is to be decided by a simple majority of the votes of its value from $2n$+1 30-second frames. Then the probability of getting the wrong result for any given ephemeris bit is

$$p_{errn} = \sum_{j=0}^{n} \frac{(1-p_{err})^j \, p_{err}^{(2n+1-j)} (2n+1)!}{j!\,(2n+1-j)!} \qquad (32)$$

Suppose that $2n$+1 = 21 frames, which equals 630 seconds of tracking data, and that the bit error rate is p_{err} = 0.14, as in the previously mentioned 15 dB Hz case. Then the probability that 21 votes will yield a majority for the wrong bit sign is $p_{errn} = 3.6\times10^{-5}$, and the probability that all of the bits in a 30-second frame will be "elected" to the correct value is 0.95. Thus, 630 seconds of data will be enough to correctly decode the ephemeris in 19 out of 20 cases.

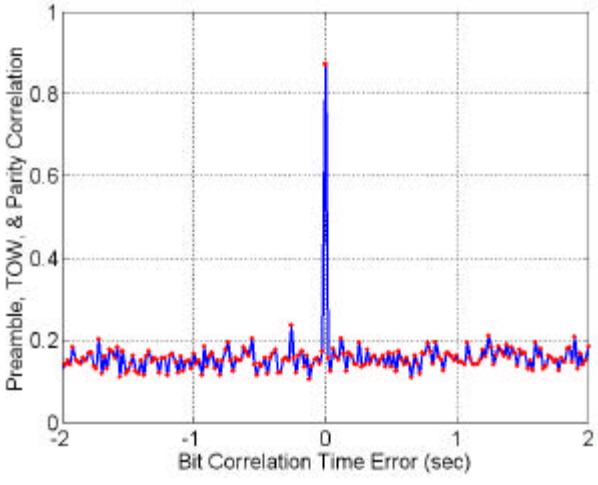

Fig. 6. Correlation of known sub-frame marker bits with tracker's bit estimates, using N_b = 1, the TCXO, and C/N_0 = 18 dB Hz, and summing absolute values over 23 sub-frames.

The Kalman filter tracking algorithm lends itself to partial bit aiding. If one knows some of the bits, then one can assign the values 1 or 0 to the a priori probabilities $p_m^{(+)}$ and $p_m^{(-)}$. The sub-frame preamble, TOW, and 2nd-word parity bits are known once sub-frame lock has occurred. In addition, many of the almanac bits and the high bits of the ephemeris data can be known ahead of time because they do not change often. The details of how best to use such information remains to be worked out, but the current tracking system should provide a means for exploiting it. The bottom plot of Fig. 4 shows that the EKF is cognizant of the possibility of a half-cycle slip whenever one occurs if it uses N_b = 4. The sign of the correlation with the sub-frame preamble, the TOW, and the 2nd-word parity bits could be used to determine whether a half-cycle slip had occurred, and that information could be used to go back and re-filter the data in order to correct the cycle slips. Re-filtering

can be accomplished rapidly using stored values of the 50 Hz *I* and *Q* accumulations along with the carrier NCO's phase.

VI. SUMMARY AND CONCLUSIONS

Fine acquisition and tracking algorithms have been developed for the carrier phase, the carrier amplitude and the code phase of a weak GPS L1 C/A signal. The fine acquisition algorithm is needed in order to provide initial signal phase and amplitude estimates with sufficient accuracy to allow the extended Kalman filter tracking algorithm to achieve lock. The fine acquisition starts with rough estimates of the carrier Doppler shift and the code phase. It refines them by solving a sequence of batch maximum likelihood estimation problems that are defined based on a time series of in-phase and quadrature accumulations. It also estimates carrier phase and the initial PRN code period of a navigation data bit. The tracking algorithm implements a combined PLL/DLL by using iterated extended Kalman filtering techniques. These techniques recursively optimize a fit between 50 Hz *I* and *Q* accumulations and models of these quantities that include sines and cosines of carrier phase errors and correlation functions evaluated at code phase errors. The Kalman filter includes a third-order carrier phase dynamic model that approximates the effects of line-of-sight acceleration and receiver clock drift. Its code phase dynamic model includes carrier aiding and random-walk code/carrier divergence. The Kalman filter uses a special Bayesian analysis of navigation data bit signs that develops alternate signal state estimates for different assumptions about a bit's sign. It estimates a posteriori probabilities for each sign and uses these probabilities to mix the different state estimates into an optimal reconstruction of the signal.

The fine acquisition algorithm and the EKF tracking algorithm can successfully acquire and track very weak GPS signals. Signals with carrier-to-noise densities as low as 22 dB Hz can be tracked with full carrier and code lock when the receiver clock is a temperature compensated crystal oscillator. Carrier cycle slips develop as C/N_0 drops below 22 dB Hz, and code and Doppler lock are lost at 15 dB Hz. There are several avenues by which improvements can be made in these thresholds. If the algorithm applies its Bayesian analysis to the 4 most recent data bits, then the full-lock threshold decreases to 20 dB Hz. If the temperature compensated crystal oscillator is replaced by an ovenized crystal oscillator and if the dynamic motion of the user vehicle is benign, as is the case in geostationary Earth orbit, then full lock can be maintained at C/N_0 = 15 dB Hz.

ACKNOWLEDGMENTS

This work has been support in part by NASA Goddard Space Flight Center through cooperative agreement no. NCC5-563 and by the NASA Office of Space Science through grant No. NAG5-12211. Luke Winternitz is the monitor for the cooperative agreement, and James Spann is the monitor for the grant.

REFERENCES

1. Spilker, J.J. Jr., *Digital Communications by Satellite*, Prentice-Hall, (Englewood Cliffs, N.J., 1977), pp. 336- 397, 446-447, & 528-608.
2. 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.
3. 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," *Proc. of ION 55th Annual Meeting*, June 28-30, 1999, Cambridge, MA, pp. 653-665.
4. 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, *Proc. of 15th Intn'l. Symposium on Spaceflight Dynamics*, CNES, June 26-30, 2000, Biarritz, France, pp. unnumbered.
5. Moreau, M.C., Davis, E.P., Carpenter, J.R., Davis, G.W., and Axelrad, P., "Results from the GPS Flight Experiment on the High Earth Orbit AMSAT-OSCAR 40 Spacecraft," *Proc. of ION GPS*, Sept. 24-27, 2002, Portland, OR.
6. Doherty, P.H., Delay, S.H., Valladares, C.E., and Klobuchar, J.A., "Ionospheric Scintillation Effects in the Equatorial and Auroral Regions," *Proc. of ION GPS*, Salt Lake City, UT, Sept. 19-22, 2000, pp. 662-671.
7. Psiaki, M.L., 'Block Acquisition of Weak GPS Signals in a Software Receiver," *Proc. of ION GPS*, Salt Lake City, UT, Sept. 11-14, 2001, pp. 2838-2850.
8. Psiaki, M.L., "Smoother-Based GPS Signal Tracking in a Software Receiver," *Proc. of ION GPS*, Salt Lake City, UT, Sept. 11-14, 2001, pp. 2900-2913.
9. Brown, R.G. and Hwang, P.Y.C., *Introduction to Random Signals and Applied Kalman Filtering, 3rd Edition*, J. Wiley & Sons, (New York, 1997), pp. 428-432.
10. Gill, P.E., Murray, W., and Wright, M.H., *Practical Optimization*, Academic Press, (New York, 1981), pp. 88-93, 99-114, & 133-140.
11. Bierman, G.J., *Factorization Methods for Discrete Sequential Estimation*, Academic Press, (New York, 1977), pp. 69-76, 115-122.
12. Psiaki, M.L., Theiler, J., Bloch, J., Ryan, S., Dill, R.W., and Warner, R.E., "ALEXIS Spacecraft Attitude Reconstruction with Thermal/Flexible Motions Due to Launch Damage," *Journal of Guidance, Control, and Dynamics*, Vol. 20(5), 1997, pp. 1033-1041.