GNSS Signal Tracking Using a Bank of Correlators

Karen Q.Z. Chiang, *Cornell University*
Mark L. Psiaki, *Cornell University*

BIOGRAPHIES

Karen Q.Z. Chiang received a B.S. in Applied Physics from Columbia University in 2009. She is currently a second-year Ph.D. student in the Sibley School of Mechanical and Aerospace Engineering at Cornell University. Her area of interest is GNSS applications for model-based estimation.

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

ABSTRACT

A combined PLL/DLL algorithm is developed for tracking GNSS carrier phase and code phase using the output from a large number of correlators. This approach has advantages for limb-scanning applications, in which useful meteorological information, available only at the initial rising time of a GPS satellite, is desired. The technique uses the bank of correlators to span wide ranges of uncertainty in code phase and carrier phase, thereby avoiding the need for a separate acquisition and the associated loss of an initial span of data. A fusion of optimal estimation methods processes the output from these correlators. A batch optimization of a signal model's fit to the many accumulations provides the Kalman filter with "measurements" of the most likely signal parameters, and the Kalman filter utilizes a signal dynamics model to provide estimates that drive the PLL and DLL.

The effectiveness of this algorithm is demonstrated by using a truth-model simulation of a limb scan. With this method and 50 Hz accumulations, pull-ins of at minimum 122 Hz and 5 C/A code chips, for the PLL and DLL respectively, are achieved. Moreover, uncertainties and errors for all signal parameters are rapidly driven close to zero. The rising signal is tracked successfully from approximately 0.03 s since emergence.

I. INTRODUCTION

GNSS receivers must achieve and maintain lock on carrier Doppler shift and the pseudo-random number (PRN) code phase in order to properly track a signal and ascertain navigation observables. A standard GNSS receiver accomplishes this with two separate, consecutive operations: acquisition and tracking [1]. Acquisition searches for initial estimates of Doppler shift and PRN code phase. The tracking algorithm then uses these estimates to initiate a delay-locked loop (DLL), and either a frequency-locked loop (FLL) or a phase-locked loop (PLL), to respectively keep the replicas of the code and carrier signal aligned with those of the received signal.

The objective of this paper is to create a joint PLL/DLL algorithm that functions normally even with large tracking errors, and that does not require the usual transition from initial acquisition to tracking. The primary motivation for such an algorithm is to equip a LEO satellite, acting as a GPS receiver, with the means to capture data from a rising-GPS-satellite limb scan, without any loss of data during the time it would take to carry out a standard acquisition.

To realize this goal of robustness and speed, the new tracking algorithm utilizes a bank of correlators to encompass uncertainties in code phase and Doppler shift, forming rectangular regions within Doppler-shift/code-phase space. Figure 1 illustrates this concept. Note that ω_D indicates the Doppler shift axis, and t_s indicates the PRN code delay axis.

In the space of these two signal properties, with time as a parameter, the estimated trajectory begins at the first estimate of Doppler shift and code phase. The uncertainty at this time is relatively large, but the bank of correlators spans a range that contains the true point. As time progresses, the level of uncertainty may change. In the case depicted, the level shrinks, and therefore the size of correlator bank will also decrease.

 Preprint from ION GNSS 2010

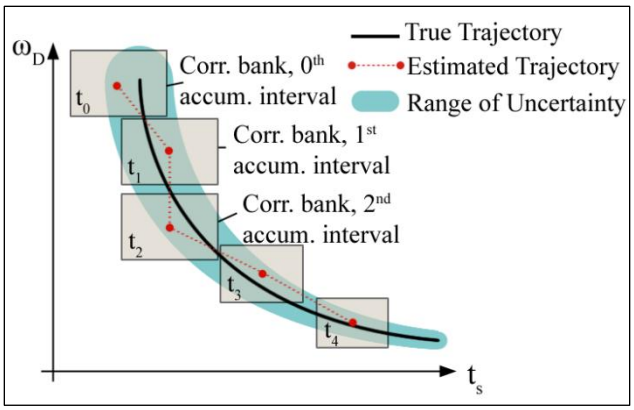

Fig. 1. Using correlator banks to cover the true trajectory and range of uncertainty in Doppler-shift/code-phase space, an illustration.

This paper's method seeks to continuously track time-varying carrier phase, carrier Doppler shift, carrier amplitude and code phase. Each bank of correlators, however, is similar to that of a brute-force acquisition, although possibly over a smaller range of uncertainty. This bank does not generate continuous numerically controlled oscillator (NCO) replicas of code and carrier phase. One of the paper's contributions consists of a way to use these correlator banks as though they produced continuous carrier and code NCO phases.

The accumulation measurements, produced by the bank of correlators, are handled by optimal estimation techniques: batch nonlinear optimization and Kalman filtering. Batch estimation, in this case the Levenberg-Marquardt method, acts as a measurement pre-processor to the Kalman filter, as shown in Figure 2. One could interpret its pre-processing calculations as a special coupled carrier-phase/carrier-Doppler-shift/code-phase discriminator.

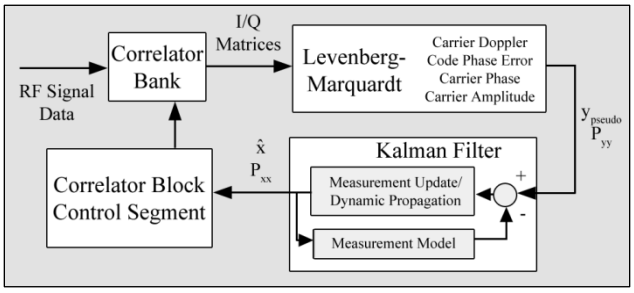

Fig. 2. Block diagram of tracking algorithm, showing pre-processing behaviour of Levenberg-Marquardt batch estimation.

Batch nonlinear optimization yields "pseudo-measurements" that the Kalman filter uses in the measurement update, while the main loop fulfills the ultimate purpose, which is tracking the estimates of carrier phase, code phase and carrier amplitude that constitute the states of the dynamic model.

The new algorithm has been evaluated by using measurements from a truth-model simulation of limb-scanning. These measurements simulate the accumulations from a given correlator bank, based on the carrier Doppler shift and code phase offsets between the true limb-scanning signal and each element of the correlator bank.

This paper begins by reviewing the signal model in Section II, from which the accumulation measurement model is derived in Section III. It then gives the dynamic model for carrier phase, code phase and carrier amplitude in Section IV. Sections V, VI and VII are devoted to the implementations of batch nonlinear optimization, the Kalman filter, and the PLL and DLL feedback control laws. Lastly, in Sections VIII and IX, the paper explains the mechanics of the limb scan simulation and analyzes the results of applying the algorithm to the simulation.

II. SIGNAL MODEL

Batch optimization and Kalman filter algorithms require a model of how carrier phase, carrier Doppler shift, carrier amplitude and code phase are related to the correlations that are computed in a receiver. A correct accumulation model starts with a signal model that describes the output of the RF front-end. This model assumes that the receiver RF front-end produces an intermediate-frequency output signal in the form

$$y(\tau_i) = Ad_k C\left[\left(\tau_i - t_{s_k}\right)(1+\eta)\right] \cdot \cos\left[\omega_{IF}\tau_i + \phi_k + \left(\tau_i - t_{DLL_k}\right)\omega_{Davg_k}\right] + n_i \quad (1)$$

where A is the signal amplitude, d_k is the 50 Hz GPS navigation data stream of ±1 values, $C(t)$ is the PRN code of the received signal, t_{s_k} is the PRN code start time, $\eta = \omega_{Dk}/\omega_{carr}$ is the non-dimensional Doppler on the PRN code chipping rate, ω_{IF} is the nominal intermediate frequency, the frequency to which the RF front-end mixes the nominal carrier frequency ω_{carr}, ϕ_k is the k^{th} (negative) beat carrier phase, t_{DLL_k} is start time of the k^{th} accumulation interval, ω_{Davg_k} is the average Doppler shift during the k^{th} accumulation interval, and n_i is a sample of zero-mean discrete-time Gaussian white noise with variance σ_n^2. The carrier-to-noise density of the sampled signal is $C/N_0 = A/(4\sigma_n^2\delta\tau)$, where $\delta\tau = \tau_{i+1} - \tau_i$ is the RF front-end sampling interval. For the purposes of the paper, the navigation bit d_k is assumed to be known at all times. In addition, note that ϕ_k is the time integral of the carrier Doppler shift, which makes it the negative of the usual definition of beat carrier phase. Also note that the DLL attempts to keep t_{DLL_k} close to t_{s_k}. The DLL keeps track of t_{DLL_k} as part of its strategy for providing a continuous record of its code phase estimates.

III. ACCUMULATION MEASUREMENT MODELS

The receiver accumulates correlations between $y(\tau_i)$ and replicas of carrier and code signals. The recipes for its in-phase and quadrature accumulations take the form

$$I_k^{l,p} = \sum_{i=i_{k0}}^{i_{k0}+N_k-1} y(\tau_i)\, C\left[\left(\tau_i - \Delta t_{DLL_{p,k}} - t_{DLL_k}\right)\right] \cdot \cos\left[\left(\omega_{IF} + \omega_{PLL_{l,k}}\right)(i - i_{k0})\delta\tau_s + \phi_{l0k}\right] \quad (2a)$$

$$Q_k^{l,p} = \sum_{i=i_{k0}}^{i_{k0}+N_k-1} y(\tau_i)\, C\left[\left(\tau_i - \Delta t_{DLL_{p,k}} - t_{DLL_k}\right)\right] \cdot \sin\left[\left(\omega_{IF} + \omega_{PLL_{l,k}}\right)(i - i_{k0})\delta\tau_s + \phi_{l0k}\right] \quad (2b)$$

where the ranges of Doppler shifts and code delays that define the correlator bank are the following:

$$\omega_{PLL_{l,k}} = \left[l - \frac{1}{2}(L-1)\right]\Delta\omega_{PLL} + \omega_{PLLavg_k} \quad for\ l = 0, \ldots, L-1 \quad (3a)$$

$$\Delta t_{DLL_{p,k}} = \left[p - \frac{1}{2}(P-1)\right]\Delta t_{DLL} + \Delta t_{DLLavg_k} \quad for\ l = 0, \ldots, L-1 \quad (3b)$$

where $\Delta\omega_{PLL}$ and Δt_{DLL} are the carrier Doppler shift and code delay spacings of the correlator bank, and ω_{PLLavg_k} and Δt_{DLLavg_k} are, respectively, the predicted carrier Doppler shift and code delay error for the interval. i_{k0} is the minimum i such that $t_{DLL_k} \le \tau_i$, N_k is the maximum N such that $\tau_{ik0+N-1} < t_{DLL_{k+1}}$, and ϕ_{l0k} is the initial intermediate-frequency phase offset of the baseband mixing signal for the particular correlator and accumulation interval.

This model is different from traditional continuous-phase carrier NCO's, especially given that there are multiple NCO Doppler shifts. Equations (2a) and (2b) constitute recipes that will actually be implemented in an FPGA, or some other digital hardware, to calculate the accumulations for its bank of correlators. However, the above model is needed for relating these accumulations to the signal parameters that the estimation methods will determine.

The model below is used to design estimators that deduce carrier phase, code phase and carrier amplitude from the accumulation outputs of the bank of correlators. It has been constructed by substituting Eq. (1) into Eq.'s (2a) and (2b), and by using trigonometric product identities, by assuming that the summation will filter out frequencies near $2\omega_{IF}$, and by using approximations of nearly continuous-time sampling and large N_k. The final measurement model takes the form

$$\begin{aligned}\boldsymbol{h_k^{l,p}} &= \begin{pmatrix} I_k^{l,p} \\ Q_k^{l,p} \end{pmatrix} \\ &= \frac{AN_k}{2} d_k \begin{pmatrix} \cos\left(\phi_{PLL_{l,k}} - \phi_{IF_k} - \phi_{avg_k}\right) \\ \sin\left(\phi_{PLL_{l,k}} - \phi_{IF_k} - \phi_{avg_k}\right) \end{pmatrix} \\ &\quad \cdot sinc\left[\left(\omega_{PLL_{l,k}} - \omega_{Davg_k}\right)\delta t_k\right] \\ &\quad \cdot sinc\left[\left(\omega_{PLL_{l,k}} - \omega_{Davg_k}\right)\delta t_k\right] \\ &\quad \cdot R\left(\Delta t_{smid_k} - \Delta t_{DLL_{p,k}}\right) \\ &\quad \cdot R\left(\Delta t_{smid_k} - \Delta t_{DLL_{p,k}}\right) \\ &\quad + \begin{pmatrix} n_{I_k}^{l,p} \\ n_{Q_k}^{l,p} \end{pmatrix}\end{aligned} \quad (4)$$

where $\phi_{PLL_{l,k}} = \left(\omega_{IF} + \omega_{PLL_{l,k}}\right)\left(\frac{N_k-1}{2}\right)\Delta\tau_s + \phi_{l0k}$ is the total Doppler-shifted mixing signals' intermediate-frequency phase at the midpoint of the accumulation samples, $\phi_{IF_k} = mod\left\{\omega_{IF}\left[\tau_{ik0} + \left(\frac{N_k-1}{2}\right)\Delta\tau_s\right], 2\pi\right\}$ is the common-mode phase associated with the nominal intermediate frequency at the midpoint, ϕ_{avg_k} is average carrier phase over the accumulation interval, and $\Delta t_{smid_k} = \frac{1}{2}\left(\Delta t_{s_k} + \Delta t_{s_{k+1}}\right)$ is the code phase error at the midpoint of the accumulation interval. $\Delta t_{s_k} = t_{s_k} - t_{DLL_k}$ is the code phase error at the start of the interval. If the accumulation interval is defined as $\delta t_{DLL_k} = t_{DLL_{k+1}} - t_{DLL_k}$, then $\delta t_k = \left(\delta t_{DLL_k} - \delta\tau_s\right)/2$, in the sinc functions of Eq. (4). $R(t)$ is the autocorrelation function of the PRN code. It is modelled with cubic splines at its slope discontinuities, in order to make its derivatives continuous and also take into account the actual rounding of the function's sharp corners due to the limited bandwidth of the RF front-end.

The vector $\boldsymbol{h_k^{l,p}}$ has two elements, but the correlator bank produces *P×L* such vectors. This entire set of correlation measurements can be stacked into the *2PL×1* vector

$$\boldsymbol{h_k} = \begin{pmatrix} \boldsymbol{h_k^{1,1}} \\ \boldsymbol{h_k^{1,2}} \\ \vdots \\ \boldsymbol{h_k^{1,P}} \\ \boldsymbol{h_k^{2,1}} \\ \vdots \\ \boldsymbol{h_k^{L,P}} \end{pmatrix} \quad (5)$$

Similarly, the noise terms at the end of Eq. (4) can be stacked into the *2PL×1* noise vector $\boldsymbol{n_k}$. The zero-mean, Gaussian discrete-time noise vector $\boldsymbol{n_k}$ is characterized by its *2PL×2PL* noise covariance matrix, R_k. The necessary formulas for its elements are

$$E\left[\left(n_{I_k}^{l,p}\right)\left(n_{I_k}^{j,q}\right)\right] = E\left[\left(n_{Q_k}^{l,p}\right)\left(n_{Q_k}^{j,q}\right)\right]$$
$$= \frac{\sigma_n^2}{2} N_k \cos\left(\phi_{PLL_{l,k}} - \phi_{PLL_{j,k}}\right)$$
$$\cdot sinc\left[\left(\omega_{PLL_{l,k}} - \omega_{PLL_{j,k}}\right)\left(\frac{N_k - 1}{2}\right)\delta\tau_s\right] \quad (6a)$$
$$\cdot R\left(\Delta t_{DLL_{p,k}} - \Delta t_{DLL_{q,k}}\right)$$

$$E\left[\left(n_{I_k}^{l,p}\right)\left(n_{Q_k}^{j,q}\right)\right] = -\frac{\sigma_n^2}{2} N_k$$
$$\cdot \sin\left(\phi_{PLL_{l,k}} - \phi_{PLL_{j,k}}\right)$$
$$\cdot sinc\left[\left(\omega_{PLL_{l,k}} - \omega_{PLL_{j,k}}\right)\left(\frac{N_k - 1}{2}\right)\delta\tau_s\right] \quad (6b)$$
$$\cdot R\left(\Delta t_{DLL_{p,k}} - \Delta t_{DLL_{q,k}}\right)$$

It is important to note that although optimal estimation works best when the measurement model refers directly to the raw measurements and their errors, as in Eq. (1), in this case such an approach would be inefficient. The final measurement model in Eq. (4) and the related covariance matrix described in Eq.'s (6a) and (6b) retains most of the significant signal information, if the correlator bank carrier Doppler shifts and code delays are chosen wisely.

IV. CARRIER PHASE, CODE PHASE AND CARRIER AMPLITUDE DYNAMICS

The dynamics model for carrier phase assumes the form of three cascaded integrators driven by white nose:

$$\begin{pmatrix} \phi \\ \omega_D \\ \alpha \end{pmatrix}_{k+1} = \begin{pmatrix} 1 & \delta t_{DLL_k} & \frac{1}{2}\delta t_{DLL_k}^2 \\ 0 & 1 & \delta t_{DLL_k} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \phi \\ \omega_D \\ \alpha \end{pmatrix}_k + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} w_{\phi_k} \quad (7)$$

where $(\phi, \omega_D, \alpha)^T{}_k$ are the states carrier phase, carrier Doppler shift, and rate of change of carrier Doppler shift at the start of the k[th] accumulation interval, or in other words, at time t_{DLL_k}. Recall that δt_{DLL_k} is the length of the k[th] accumulation interval. w_{ϕ_k}, another zero-mean, discrete-time Gaussian noise sequence, is the carrier phase process noise.

The states of this linear system can be used to derive the average carrier phase over the accumulation interval between times t_{DLL_k} and $t_{DLL_{k+1}}$ [2]:

$$\phi_{avg_k} = \begin{pmatrix} 1 & \frac{1}{2}\delta t_{DLL_k} & \frac{1}{6}\delta t_{DLL_k}^2 \end{pmatrix} \begin{pmatrix} \phi \\ \omega_D \\ \alpha \end{pmatrix}_k + (0 \quad 0 \quad 0 \quad 1) w_{\phi_k} \quad (8)$$

This is the phase that is subtracted from the NCO phase in the measurement model expressed in Eq. (4). Similarly, the average Doppler shift over the accumulation interval is

$$\omega_{D_{avg_k}} = \begin{pmatrix} 0 & 1 & \frac{1}{2}\delta t_{DLL_k} \end{pmatrix} \begin{pmatrix} \phi \\ \omega_D \\ \alpha \end{pmatrix}_k + \begin{pmatrix} \frac{1}{\delta t_{DLL_k}} & 0 & 0 & 0 \end{pmatrix} w_{\phi_k} \quad (9)$$

This is the Doppler shift that is subtracted from the NCO Doppler shift in the measurement model of Eq (4).

The noise covariance matrix associated with the white process noise in this dynamic model takes into account the random walk acceleration of the line-of-sight (LOS) vector, as well as the random walks of the receiver clock frequency and receiver clock phase. The covariance matrix for $w_{\phi_k}$is

$$E\left(w_{\phi_k} w_{\phi_k}^T\right)$$
$$= q_{LOS} \begin{pmatrix} \frac{1}{20}\delta t_{DLL_k}^5 & \frac{1}{8}\delta t_{DLL_k}^4 & \frac{1}{6}\delta t_{DLL_k}^3 & \frac{1}{72}\delta t_{DLL_k}^5 \\ \frac{1}{8}\delta t_{DLL_k}^4 & \frac{1}{3}\delta t_{DLL_k}^3 & \frac{1}{2}\delta t_{DLL_k}^2 & \frac{1}{30}\delta t_{DLL_k}^4 \\ \frac{1}{6}\delta t_{DLL_k}^3 & \frac{1}{2}\delta t_{DLL_k}^2 & \delta t_{DLL_k} & \frac{1}{24}\delta t_{DLL_k}^3 \\ \frac{1}{72}\delta t_{DLL_k}^5 & \frac{1}{30}\delta t_{DLL_k}^4 & \frac{1}{24}\delta t_{DLL_k}^3 & \frac{1}{252}\delta t_{DLL_k}^5 \end{pmatrix}$$
$$+ S_g \omega_{carr}^2 \begin{pmatrix} \frac{1}{3}\delta t_{DLL_k}^3 & \frac{1}{2}\delta t_{DLL_k}^2 & 0 & \frac{1}{8}\delta t_{DLL_k}^3 \\ \frac{1}{2}\delta t_{DLL_k}^2 & \delta t_{DLL_k} & 0 & \frac{1}{6}\delta t_{DLL_k}^2 \\ 0 & 0 & 0 & 0 \\ \frac{1}{8}\delta t_{DLL_k}^3 & \frac{1}{6}\delta t_{DLL_k}^2 & 0 & \frac{1}{20}\delta t_{DLL_k}^3 \end{pmatrix}$$
$$+ S_f \omega_{carr}^2 \begin{pmatrix} \delta t_{DLL_k} & 0 & 0 & \frac{1}{2}\delta t_{DLL_k} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2}\delta t_{DLL_k} & 0 & 0 & \frac{1}{3}\delta t_{DLL_k} \end{pmatrix} \quad (10)$$

where q_{LOS} is the power spectral density of the continuous-time white noise that drives the acceleration random walk, S_g is the power spectral density of the white noise that drives the receiver clock frequency random walk, and S_f is the power spectral density of the white noise that drives the clock phase random walk [2].

The dynamic model for the PRN code phase keeps track of the true code start and stop times associated with the nominal PRN code segment for the accumulation interval:

$$t_{s_{k+1}} = t_{s_k} + \frac{\omega_{carr}\delta t_{nom} - [1 \quad 0 \quad 0 \quad 0]w_{\phi_k}}{\omega_{carr} + \omega_k + \frac{1}{2}\delta t_{nom}\alpha_k} + w_{ts_k} \tag{11}$$

Where $\delta t_{nom}$is the nominal length of the code segment and $w_{ts_k}$is a white noise term that models the random walk of code phase. The second term on the right-hand side of Eq. (11) is the carrier-aiding term that captures the coupling between carrier Doppler shift and code chipping rate. The dynamic model for code phase error is therefore

$$\Delta t_{sk+1} = \Delta t_{s_k} + \frac{\omega_{carr}\delta t_{nom} - [1 \quad 0 \quad 0 \quad 0]w_{\phi_k}}{\omega_{carr} + \omega_k + \frac{1}{2}\delta t_{nom}\alpha_k} - \delta t_{DLL_k} + w_{ts_k} \tag{12}$$

Note that the second and third terms on the right-hand side of Eq. (12) comprise the difference between the true length of the code segment and the DLL's estimate of it. The DLL attempts to keep this difference near zero. The PRN code phase error at the midpoint of the accumulation, which is needed as part of the argument for the autocorrelation function in Eq. (4) is

$$\Delta t_{smid_k} = \Delta t_{s_k} + \frac{1}{2}\frac{\omega_{carr}\delta t_{nom} - [1 \quad 0 \quad 0 \quad 0]w_{\phi_k}}{\omega_{carr} + \omega_k + \frac{1}{2}\delta t_{nom}\alpha_k} - \frac{1}{2}\delta t_{DLL_k} + \frac{1}{2}w_{ts_k} \tag{13}$$

Also modelled as a purely random walk is carrier amplitude:

$$A_{k+1} = A_k + w_{A_k} \tag{14}$$

where $w_{A_k}$is the discrete-time white process noise that drives the random walk. This equation can be used to deduce the average amplitude of the accumulations:

$$A_{IQ_k} = \frac{1}{2}N_k A_k + \frac{1}{4}N_k w_{A_k} \tag{15}$$

V. BATCH NONLINEAR OPTIMIZATION USING THE LEVENBERG-MARQUARDT METHOD

Kalman filters typically contain two stages of computation: Dynamic propagation and measurement update. Measurement updating adjusts the a priori states according to incoming measurements. Batch optimization in the present algorithm provides these measurements as a multi-correlator vector discriminator of carrier Doppler shift, code phase, carrier phase and carrier amplitude.

Specifically, batch optimization fits accumulation data coming out of the correlator bank to the measurement model in Eq. (4), and yields the most likely signal parameters associated with the best fit. These parameters are the batch states for which the Levenberg-Marquardt algorithm seeks the solution. The states are average Doppler shift, midpoint code phase error, average carrier phase and average carrier amplitude, denoted in this paper by $\left(\omega_{Davg_k}, \Delta t_{smid_k}, \phi_{avg_k}, A_{IQ_k}\right)$. The equations for these variables were given in the previous section. In essence, the output of the batch optimization algorithm provides something akin to partial measurement linearization and sensitivity adjustment of the accumulations about the optimal values of these signal parameters.

Batch nonlinear optimization starts by choosing the correlator, indexed by l, p, that has the highest $I^2 + Q^2$accumulation power, along with its nearest neighbours. This search is conducted along both the Doppler shift and code phase directions, and the calculations are the same as those in a normal GNSS acquisition search. Figure 3 shows a superposition of continuous, theoretical power, which the receiver never actually sees, and discrete, correlator-measured power.

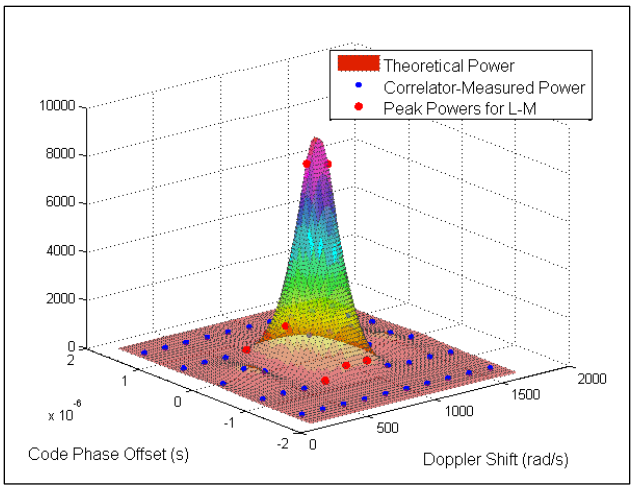

Fig. 3. Measured power of accumulations from a correlator bank, superimposed on theoretical power.

Each dot represents a correlator that mixes the RF front-end output signal with the appropriate NCO Doppler shift and code phase offset within the specified ranges. Note that the effects of noise have been neglected in generating Figure 3, thereby causing the dots to fall directly on the theoretical plot. The number of correlators used will depend on the uncertainty of these two signal parameters in the current accumulation interval. Uncertainty here is based on the Doppler shift and code phase offset

variances in the state error covariance matrix determined by the Kalman filter. The spacing of the correlator grid is predetermined in the simulation with reasonable values that are close enough to find the peak, but not so close that they begin to cause excessive computational cost and numerical conditioning problems.

The red dots depict the correlators that are chosen to have their accumulation measurements sent to the batch estimation algorithm for data fitting. A clearer view of the correlator grid and these selected points is shown in the power contour plot of Figure 4.

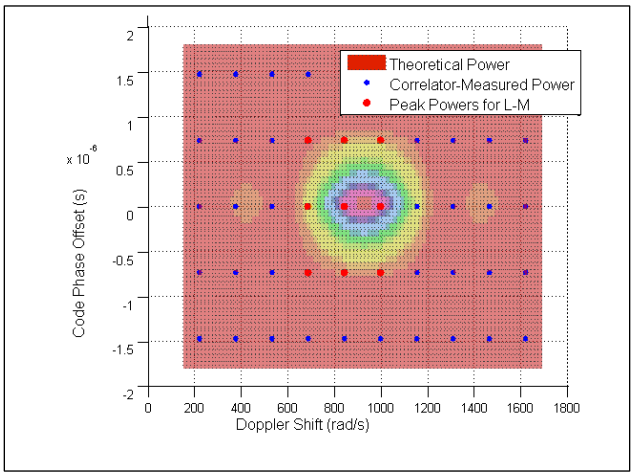

Fig. 4. Power contours mapped onto a bank of correlators.

In the plot above, it is obvious that none of the chosen correlators, indicated in red, need to be exactly on the peak of the power function. The Levenberg-Marquardt solution, however, will give the best accuracy if the selected points span the peak.

Data fitting is performed by minimizing the cost function

$$J\left(\omega_{Davg_k}, \Delta t_{smid_k}, \phi_{avg_k}, A_{IQ_k}\right) = \frac{1}{2}\left[\boldsymbol{y}_{\boldsymbol{tr_k}} - \boldsymbol{h}_{\boldsymbol{tr_k}}\left(\omega_{Davg_k}, \Delta t_{smid_k}, \phi_{avg_k}, A_{IQ_k}\right)\right]^T {R_{tr_k}}^{-1} \cdot \left[\boldsymbol{y}_{\boldsymbol{tr_k}} - \boldsymbol{h}_{\boldsymbol{tr_k}}\left(\omega_{Davg_k}, \Delta t_{smid_k}, \phi_{avg_k}, A_{IQ_k}\right)\right] \quad (16)$$

The measurement vector $\boldsymbol{y}_{\boldsymbol{tr_k}}$ is a truncated version of the vector of I's and Q's obtained directly from the correlator bank, and the vector $\boldsymbol{h}_{\boldsymbol{tr_k}}$ is the corresponding truncated version of $\boldsymbol{h}_{\boldsymbol{k}}$ from Eq. (5). This truncation eliminates the correlations associated with the blue dots in Figures 3 and 4, to retain only those associated with the red dots, i.e. the correlations that have significant power. The inverse covariance matrix ${R_{tr_k}}^{-1}$ will likewise be a smaller version of the original noise covariance matrix. The cost J is the negative log likelihood of its four input parameters. Therefore, the Levenberg-Marquardt solution is the maximum likelihood solution.

Before implementing the Levenberg-Marquardt algorithm, it is helpful to recognize that ϕ_{avg_k} and A_{IQ_k} can be optimized analytically. In order to do this, define

$$\boldsymbol{x} = (x_1 \quad x_2)^T = \left[A_{IQ_k}\cos(\phi_{avg_k}) \quad A_{IQ_k}\sin(\phi_{avg_k})\right]^T \quad (17)$$

and set the derivative of the above cost function, with respect to this vector $\boldsymbol{x}$, to zero. The resulting necessary condition is linear in $\boldsymbol{x}$, and can be solved exactly. Analytically optimized values of A_{IQ_k} and ϕ_{avg_k} can be calculated for every pair of ω_{Davg_k} and Δt_{smid_k}:

$$A^*_{IQ_k} = \|\boldsymbol{x}\| \quad (18a)$$

$$\phi^*_{avg_k} = \text{atan2}(x_2, x_1) \quad (18b)$$

The measurement model can be easily rearranged due to its structure so that this can be done. With analytic optimization, the Levenberg-Marquardt algorithm will only need to search in the two directions of Doppler shift and code phase error, instead of in four directions. This cuts down much of the computational expense.

The Levenberg-Marquardt implementation used here has two special features [3]. The first is its use of the exact cost function Hessian, rather than the approximate Hessian that uses only first derivatives of the measurement function $\boldsymbol{h}_{\boldsymbol{tr_k}}$. The second feature is a modification of how the algorithm calculates the parameter that limits the step size. In addition to the usual requirement of cost decrease, the parameter can be adjusted upward to compensate for an indefinite cost Hessian and to keep the solution within the bounds of the chosen points in Doppler-shift/code-phase space. This limitation, paired with the use of the peak power point as the first guess, tends to guarantee that the solution is the global minimum.

The final optimized batch states become the Kalman filter's measurements. These have been called "pseudo-measurements" in the introduction due to the fact that they do not correspond directly to the measurement model in Eq. (4). Instead, the Kalman filter compares these measurements to the models established in Eq.'s (9), (13), (8) and (15), respectively. There is also an error covariance matrix associated with this measurement model. It is set equal to the inverse of the Levenberg-Marquardt optimal cost Hessian. Although this is more like a Cramer-Rao lower bound, it is a reasonable choice.

Note that an alternate method would be to apply a nonlinear Kalman filter directly to the accumulation measurements described in Eq. (4), but the present

method has the advantage of increased accuracy by "linearizing" about the parameters that give peak accumulation power, instead of only linearizing about the Kalman filter's a priori estimate. The Levenberg-Marquardt algorithm is also a way to transition between the large collections of accumulations produced by the correlator bank to a reasonable set of measurements that the Kalman filter can easily incorporate.

VI. IMPLEMENTATION OF KALMAN FILTER

Two Kalman filters are tested for the tracking algorithm: A nonlinear extended square root information filter (EKF) and a nonlinear unscented Kalman filter (UKF). The results from these two filters have no significant difference. The Kalman filter makes use of the dynamic models for carrier phase, carrier Doppler shift, rate of change of carrier Doppler shift, carrier amplitude and code phase error. This model is a combination of Eq.'s (7), (12) and (14), and is used in dynamic propagation. The model used for measurement update is the output of the batch optimization, as described in the previous section. Both processes of dynamic propagation and measurement update are nonlinear due to carrier-aiding being present in code phase dynamics. This nonlinearity is the only one remaining after Levenberg-Marquardt, and its weakness is likely the cause of the close similarity of results produced by the different extended filters.

Also implemented within the Kalman filter are four data tests that are used in order to decide whether or not to perform a measurement update. All four tests must be passed to perform an update. The first test accepts data only if the peak accumulation power in the correlator bank does not lie on a boundary in the Doppler shift or code delay range. The second test requires that the Levenberg-Marquardt algorithm converges in a reasonable number of iterations. The third test examines the optimal Leveberg-Marquardt cost. It should be half of a chi-squared sample from a distribution of degree $N_{tr_k} - 4$, where N_{tr_k} is the number of elements in $\boldsymbol{y}_{\boldsymbol{tr_k}}$ and in $\boldsymbol{h}_{\boldsymbol{tr_k}}$, or twice the number of selected red dots in Figures 3 and 4. If the optimal Levenberg-Marquardt cost is too high, say high enough that the random probability of generating it is less than 10^{-4}, then the measurement is rejected. The fourth test performs the Kalman filter measurement update, and examines the value of the sum of the squares of the normalized innovation vector. It should be a sample from a chi-squared distribution of degree four. If the value is too high, then the measurement update is rejected. Such rejections, however, are rare, except at very low carrier-to-noise ratios.

VII. PLL AND DLL FEEDBACK CONTROL LAWS

Feedback control laws are needed in order to complete the loop shown in Figure 2, by implementing the correlator block control segment. These laws take the most recent Kalman filter a posterior state estimates to predetermine the accumulation intervals, the NCO carrier Doppler shifts, as defined in Eq. (3a), and the NCO code phase offsets, as defined in Eq. (3b). The feedback from a given accumulation interval is used to set these values two accumulation intervals forward.

The DLL feedback control law determines the k+2nd accumulation interval as follows:

$$\begin{aligned}\delta t_{DLL_{k+2}} = {} & \Delta\hat{t}_{s_{k+1}} \\ & + \frac{\omega_{carr}\delta t_{nom}}{\omega_{carr} + \hat{\omega}_{D_{k+1}} + 0.5\delta t_{nom}\hat{\alpha}_{k+1}} \\ & + \frac{\omega_{carr}\delta t_{nom}}{\omega_{carr} + \hat{\omega}_{D_{k+1}} + 1.5\delta t_{nom}\hat{\alpha}_{k+1}} \\ & - \delta t_{DLL_{k+1}}\end{aligned} \tag{19}$$

The effect of this control law is to use the code phase offset estimate at $t_{DLL_{k+1}}$, which is $\Delta\hat{t}_{s_{k+1}}$, and the predicted lengths of the k+1st and k+2nd intervals, so that the predicted code phase offset at $t_{DLL_{k+3}}$ will be zero.

The DLL also needs to predict the expected average code phase error for the k+2nd accumulation interval. Although the expected error is zero at the end of the interval, as per the design of Eq. (19), it is not necessarily zero at the beginning. In fact, it equals half the sum of the first, second and fourth terms on the right-hand side of Eq. (19):

$$\begin{aligned}\Delta t_{DLLavg_{k+2}} = \frac{1}{2}\Bigg(& \Delta\hat{t}_{s_{k+1}} \\ & + \frac{\omega_{carr}\delta t_{nom}}{\omega_{carr} + \hat{\omega}_{D_{k+1}} + 0.5\delta t_{nom}\hat{\alpha}_{k+1}} \\ & - \delta t_{DLL_{k+1}}\Bigg)\end{aligned} \tag{20}$$

Eq. (3b) also requires a choice of the PRN code spacing Δt_{DLL} and the number of code offsets *P*. $\Delta t_{DLL}$is normally chosen in the range 0.5 to 1 code chip lengths. *P* is normally chosen so that $P\Delta t_{DLL}$ is four to six times the Kalman filter's 1-σ code phase uncertainty, as computed from its covariance matrix. Note that a minimum value of *P = 3* is enforced when the code delay uncertainty is very small.

The nominal Doppler shift predicted by the PLL for the k+2nd accumulation interval is

$$\omega_{NCOavg_{k+2}} = \hat{\omega}_{D_{k+1}} + \left(\delta t_{DLL_{k+1}} + \frac{1}{2}\,\delta t_{DLL_{k+2}}\right)\hat{\alpha}_{k+1} \quad (21)$$

Eq. (3a) also requires a choice of the Doppler shift spacing $\Delta\omega_{PLL}$and the number of Doppler offsets L. $\Delta\omega_{PLL}$ is normally chosen in the range $\pi/\delta t_{nom}$ to $2\pi/\delta t_{nom}$, i.e. from ½ to 1 times the accumulation frequency. L is normally chosen so that $L\Delta\omega_{PLL}$is four to six times the Kalman filter's 1-σ Doppler shift uncertainty. Note, however, that a minimum value of $L = 3$ is always used, even when the Doppler shift uncertainty is very small.

The "hat" accents denote estimates given by the Kalman filter, for time $t_{DLL_{k+1}}$. These do not become available until shortly after that time, because they are based on accumulations from the interval that ends at that time. Presumably, the receiver's processor can finish computing all of these feedback quantities during the interval from $t_{DLL_{k+1}}$to $t_{DLL_{k+2}}$, so that they will be available to the correlator block control segment during the interval from $t_{DLL_{k+2}}$ to $t_{DLL_{k+3}} = t_{DLL_{k+2}} + \delta t_{DLL_{k+2}}$.

VIII. LIMB SCAN SIMULATION

The rising-GPS-satellite limb-scanning application, with a low-Earth orbit (LEO) satellite carrying a GPS receiver, is an ideal application for this acquisitionless tracking system. The most useful meteorological data occurs just at the initial point when the satellite rises, but it is difficult to start tracking the signal immediately at this point with traditional receiver algorithms. An acquisition would waste some of this valuable data.

To investigate the new system's performance on this problem, a high-fidelity truth-model simulation of a limb scan has been developed. Its goal is to show that the new Kalman Filter can start tracking the signal with high accuracy on its carrier and code phase, immediately upon availability. Thus, meteorological data could be captured for limb scans with minimum altitudes of only several meters.

The truth-model simulation includes LEO orbital dynamics, namely, a circular orbit with 700 km of altitude and 98.2° of inclination. The GPS satellite orbital dynamics are modelled using standard GPS ephemerides and orbit calculations. As the orbital dynamics cause the signal to penetrate the atmosphere, the signal path experiences refraction dictated by the generalized 3D Snell's law differential equation, and through modelled neutral atmosphere density and ionosphere electron density distributions. The solution for the refracted path is that of a two-point boundary value problem (TPBVP) between each LEO and GPS satellite location. Figure 5 renders an example of a curve determined by the TPBVP solver. The plot shows the geometry of the Earth limb (blue), the straight-line path (green) which is occulted, and the refracted bent path (red) which has a minimum altitude of 54 m.

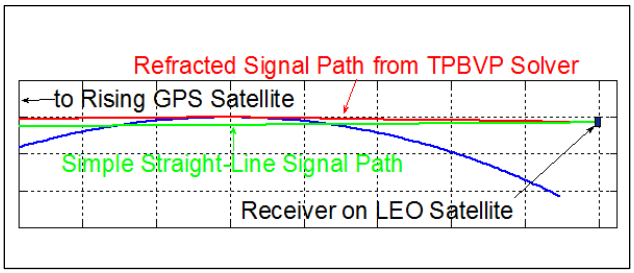

Fig. 5. Sketch of limb-scan geometry.

The effects on carrier Doppler shift and pseudorange are calculated at discrete time points and then interpolated between these times. These effects are due to tropospheric and ionospheric refraction, and cause large initial perturbations in pseudorange and Doppler shift, as shown in Figure 6. The modelled refraction includes the phenomena of geometric path bending, signal group-delay/phase-delay in the troposphere, and group-delay/phase-advance in the ionosphere.

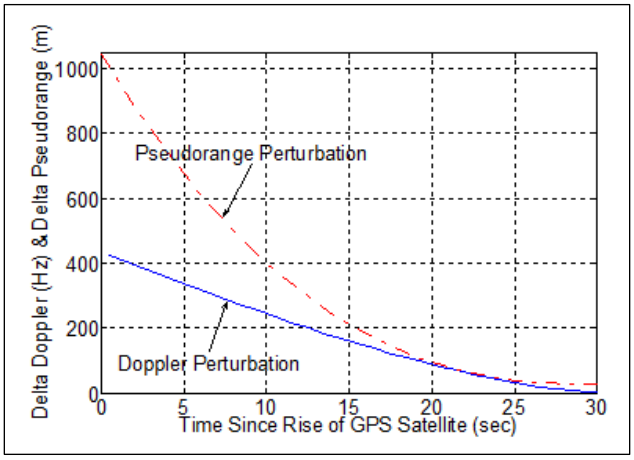

Fig. 6. Pseudorange and carrier Doppler shift perturbation time histories due to refraction.

The change in Doppler shift in the first 20 seconds is about 300 Hz, translating into an apparent unmodelled acceleration of about 0.3 g at the L1 carrier frequency. The initial pseudorange offset is 1040 m, which implies a code phase uncertainty of about 3.5 chips of C/A code.

The effects of receiver clock error are also simulated using a standard Markov model. They are given in Ref. 4.

IX. SIMULATION TESTING RESULTS

The following are results of applying the new algorithm to the limb-scan truth-model simulation. The overall simulation has been implemented in MATLAB and uses

correlator grid spaces of $\Delta\omega_{PLL} = 37.5$ Hz in Doppler shift and $\Delta t_{DLL} = 0.75$ chips in code phase offset. These are reasonable, albeit ad-hoc, values that are large enough to avoid numerical issues such as poor conditioning of noise covariance matrices. The receiver clock model uses the parameters $h_0 = 10^{-22}$ s and $h_{-2} = 7.6\times10^{-24}$ s^{-1}. These clock parameters yield a minimum root Allen variance of 10^{-11} at a delay of 1 s, consistent with an ovenized crystal oscillator. q_{LOS} is set to 140 rad^2/s^5; this value is based on measures of the degree of error between "truth" phase and the rate at which it deviates from a quadratic in the limb-scan time histories. The nominal model accumulation interval $\delta t_{nom}$is set to 0.02 s, giving 50 Hz accumulations. The carrier frequency ω_{carr} corresponds to that of the L1 signal.

Figure 7 plots the simulated time history of carrier Doppler shift, from the point that the GPS satellite rises above the Earth limb, as viewed by the LEO satellite. It is important to consider not only the speed with which the PLL achieves lock, but also its successful pull-in from an initial Doppler shift offset of 122 Hz, despite using 50 Hz accumulations. These conditions would prevent a traditional 2- or 3-correlator from ever achieving lock.

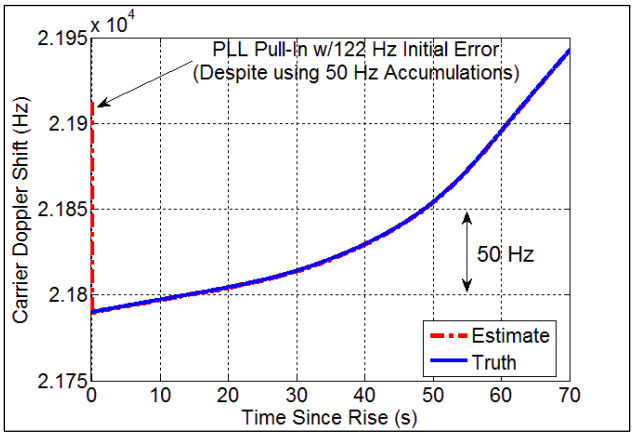

Fig. 7. Time history of carrier Doppler shift.

Similar results are seen for PRN code phase error, shown in Figure 8. The DLL also achieves and maintains lock very quickly, even with an initial code phase error of 5 chips. Uncertainty is plotted in the form of a 1-σ value, which also drops nearly to zero after only about 0.03 s.

The simulated carrier-to noise ratio and its estimate are shown in Figure 9. The signal starts with about 15 dB of attenuation due to atmospheric path loss and the effects of refractive lensing. This power loss disappears after 60 s, by which time the signal path is largely clear of the atmosphere. The congruence of the "truth" and estimated C/N_0 plots implies that signal amplitude is also being tracked effectively.

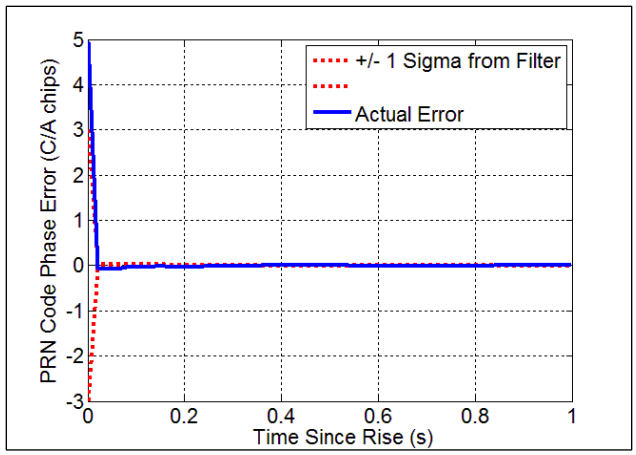

Fig. 8. Time history of PRN code phase error.

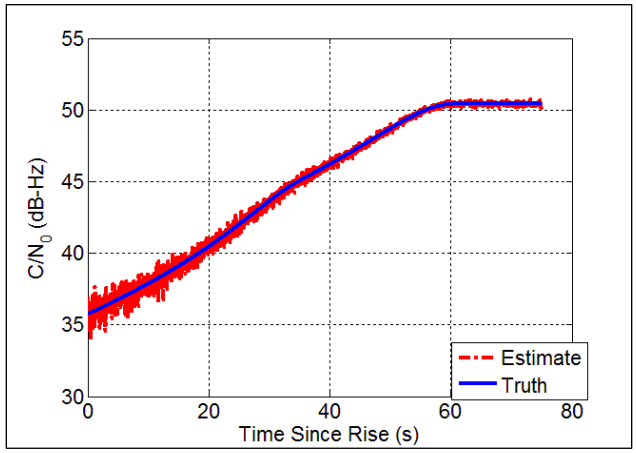

Fig. 9. Time history of carrier-to-noise ratios.

The final signal parameter of interest is carrier phase. Its error time history is graphed in Figure 10. Note once again that within about three accumulation intervals, the phase error is pulled in from a full cycle to a few degrees. The ±1-σ uncertainty is effectively infinite at filter initialization, but it decreases to a very small value within the first few accumulation intervals.

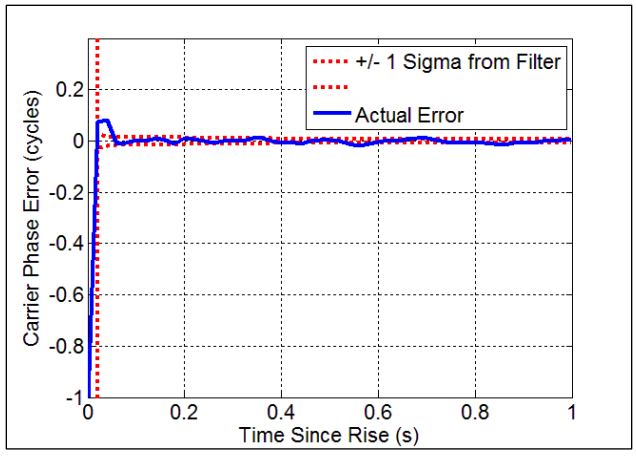

Fig. 10. Time history of carrier phase error.

The algorithm responds to the aforementioned drops in uncertainty by lowering the number of correlators that must be used, saving computational cost. Time histories of these numbers are shown in Figure 11. The bank of correlators starts with $L = 21$ NCO Doppler shifts and $P = 21$ different NCO code phase offsets. The equality of the numbers is largely a coincidence resulting from initial uncertainty ranges.

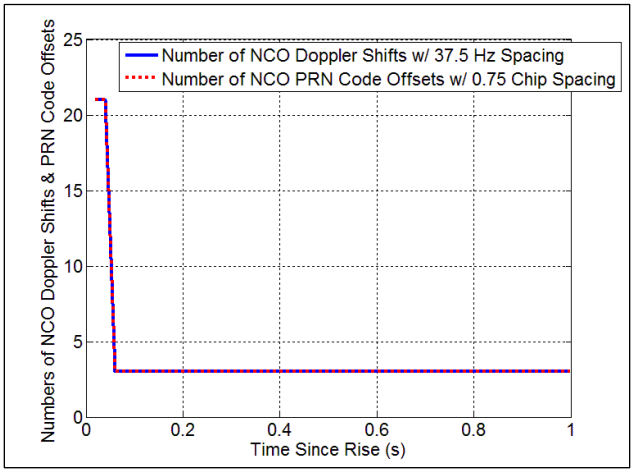

Fig. 11.Time history of correlator bank size.

The rapid reductions in uncertainty, as indicated by the Kalman filter covariance matrix, translate into drops in the numbers of NCO grid points both in the Doppler shift offset direction and in the code offset direction. After a short time, only three correlators are used in each direction to maintain lock.

CONCLUSION

A new GPS signal tracking algorithm has been developed to combine PLL and DLL functions with accumulation data from acquisition-like correlator banks. Its purpose is to achieve rapid tracking with large pull-in regions, without the need for a separate initial acquisition. The new tracking algorithm chooses the correlator with the highest accumulation power from the bank of correlators, along with some of its close neighbours in Doppler shift and code phase offset. The accumulation measurements from these selected correlators are passed onto a batch nonlinear optimization method, an intermediate step to the Kalman filter that retrieves the most likely signal parameters. The Kalman filter treats the optimized batch states as measurements and provides the PLL and DLL control laws with the necessary inputs for defining the region that must be covered by the bank of correlators.

The algorithm's use of acquisition-like data enables it to achieve robust tracking in the presence of highly dynamic signals with large initial uncertainties. For example, PLL pull-ins from 122 Hz of initial Doppler error can be achieved using 50 Hz accumulations, as can DLL pull-ins of from an initial error of 5 C/A code chips. This capability may be very useful in tracking the signal from a rising GPS satellite, received aboard a LEO platform. The resulting limb-scan data, because near-instantaneous tracking can be achieved, will provide information about meteorological conditions within a few meters of the Earth's surface.

REFERENCES

1. Misra, P., and Enge, P., *Global Positioning System: Signals, Measurements, and Performance*, Ganga-Jamuna, (Lincoln, 2006), pp. 431-492.
2. Psiaki, M.L., and Jung, H., "Extended Kalman Filter Methods for Tracking Weak GPS Signals," *Proc. ION GPS 2002*, Sept. 24-27, 2002, Portland, OR, pp. 2539-2553
3. Gill, P.E., Murray, W., and Wright, M.H., *Practical Optimization*, Academic, (London, 1981), pp. 136-137.
4. Brown, R.G., Hwang, P.Y.C., *Introduction to Random Signals and Applied Kalman Filtering*, J. Wiley, (New York, 1992), pp. 428-432.