A High Performance Code and Carrier Tracking Architecture for Ground-Based Mobile GNSS Receivers

Dr. Lawrence R. Weill, *Chief Scientist, Magellan Systems Japan*
and *Professor of Mathematics Emeritus, California State University, Fullerton*

BIOGRAPHY

Dr. Weill received B.S. and M.S. degrees in Electrical Engineering from the California Institute of Technology in 1960 and 1961, respectively. In 1968 he earned the M.S. Degree in Mathematics at San Diego State University, and was awarded the Ph.D. in Mathematics in 1974 at the University of Idaho. He is currently Chief Scientist at Magellan Systems Japan and Professor of Mathematics Emeritus from California State University, Fullerton. He has operated his own consulting firm for 31 years.

Dr. Weill is also one of the three technical founders of Magellan Systems Corporation, which in 1989 produced the world's first low-cost handheld GPS receiver for the consumer market.

As an active researcher, Dr. Weill has published numerous papers on signal processing for GNSS, radar, sonar, optical sensor, and satellite communication systems. He has made substantial contributions to both the theoretical foundations and practical aspects of GNSS signal compression and multipath mitigation, and is currently developing new approaches for high performance GNSS receivers.

ABSTRACT

It is well-known that vector delay and frequency locked loops (VDFLLs) have several advantages over traditional scalar tracking loops in GNSS receivers, especially in mobile platforms subject to poor signal environments and accelerations. This paper presents an approach to further VDFLL performance improvement for ground-based mobile receivers which apparently has not yet been exploited. The performance improvement results from using the dynamics limitations of typical ground-based vehicles, long-term pre-measurement integration, and the weak-signal advantages of direct maximum-likelihood (ML) estimation of position and velocity to materially improve weak-signal tracking capability. In the new VDFLL design, which will be denoted by MLVTL (Maximum Likelihood Vector Tracking Loop), a Kalman filter for the navigation processor is not required. Additionally, map aiding can be incorporated directly in the tracking loop to reduce position and velocity errors, lower the tracking threshold, and reduce the minimum number of satellites required to maintain tracking.

1. DESIGN PHILOSOPHY

Figure 1 is a simplified block diagram of a scalar signal tracking method which has been used for many years in GPS receivers. Each satellite signal is independently tracked with a code delay-locked loop which enables measurement of pseudorange, and a carrier frequency-locked loop which enables measurement of pseudorange rate. The measurements are fed to a navigation processor, typically a Kalman filter or recursive least-squares estimator, which produces the navigation solution for position, velocity, and time (PVT).

A simplified block diagram of a typical VDFLL tracking loop is shown in Figure 2. Predictions of position and velocity navigation states from a navigation processor (such as a Kalman filter or least-squares estimator) are converted into predictions of satellite pseudoranges and pseudorange rates, which are fed as references to code delay and carrier frequency discriminators operating on the satellite channels. These references are coupled because they all are derived from the same navigation states. The discriminators provide *measurement residuals* for each satellite, which are estimates of the difference between the predicted pseudoranges and pseudorange rates and the same parameters inherent in the received satellite signal. The measurement residuals are fed back to the navigation processor, thus closing the tracking loop. For legacy C/A coded GPS signals, the loop iteration time interval typically spans from 1 to 10 bits of the 50 bps navigation data stream (20-200 milliseconds).

The advantages of the VDFLL in Figure 2 over the scalar tracking loop in Figure 1 accrue mainly from the coupling of

the references for the code and carrier discriminators. It can be shown that this reduces tracking error due to thermal noise when the number of satellites exceeds the number of navigation states. The VDFLL is also more robust in the presence of signal dropouts and vehicle accelerations. Background information on conventional and VDFLL tracking can be found in [1-4].

However, despite the coupling of the discriminator references, it is important to note that the measurement residuals are still independently estimated for each satellite channel. This exposes a vulnerability when operating with the weak signals often encountered in mobile operation. It is well known that the measurement error from an individual discriminator increases very rapidly as the C/No drops below the level required for the error to approach the Cramer-Rao lower bound. Inevitably, this leads to outlier or "wild" measurements which disruptively propagate through the navigation processor. Since each discriminator operates in the presence of its own independent noise, there is no opportunity to increase the processing gain at this point in the loop and lower the tracking threshold by using joint signal characteristics.

To solve these problems, the new MLVTL architecture shown in Figure 3 has been developed. It does not use discriminators to provide the usual measurement residuals to the navigation processor. Instead, direct maximum-likelihood (ML) estimates of the navigation states are performed using small simultaneously-generated segments of the code correlation functions and frequency spectra from all satellites. The tracking loop is closed by using the estimated navigation states to update the centers of the segments of the code correlation functions and frequency spectra for the next ML estimate. Unlike a typical VDFLL Kalman filter, the ML estimates are repeated at a relatively slow rate (from 1 to 3 seconds apart) to permit a large amount of pre-estimation processing gain, as well as sufficient time for the ML computations. Also unlike a Kalman filter, the ML estimator is not recursive. Each estimate uses enough signal data to permit good weak-signal performance on its own.

There are other advantages. Because frequency discriminators are not used, their limited operating range (typically ± 25 Hz) is no longer a limitation on how much platform acceleration can be tolerated without loss of frequency lock, especially with weak signals. The new MLVTL design exploits the fact that in typical land-based mobile operation large accelerations (a significant fraction of 1 g) are infrequent and when they do occur, they are sustained for no more than a few seconds. Acceleration is therefore not modeled in the ML estimator. If an ML estimate of position and velocity is degraded by a large acceleration, essentially complete recovery is possible at the next estimate. Gone is the problem of making a Kalman filter properly responsive to changes in acceleration without adding more states or fiddling with its covariance matrix. Smaller accelerations simply cause a momentary small loss of sensitivity and non-noticeable position and velocity errors which do not propagate forward in time.

The new MLVTL architecture also uses altitude aiding to reduce the position and velocity dimensionality, which results in better tracking accuracy and a lower tracking threshold. Memory size for stored altitude data can be much smaller than horizontal aiding data, because altitude generally varies much more slowly than horizontal position.

2. DETAILED DESCRIPTION

The ML Estimator

In the new MLVTL, the ML estimator jointly estimates receiver position, velocity, GPS time error, and GPS time rate error. GPS time is generated within the receiver by a process that will be described later. It can be shown that the ML estimate is equivalent to a least-squares estimate which minimizes the integral

$$L \triangleq \int_{T_0}^{T_0+T} \left| r(t) - \sum_{k=1}^{N} s_k(t) \right|^2 dt \qquad (1)$$

where $r(t)$ is the complex-valued baseband received signal plus noise, $s_k(t)$ is a receiver-generated baseband replica of a noiseless signal for satellite k, N is the number of satellites observed, T_0 is the beginning of the captured signal, T is the duration of the captured signal, and t is GPS time as generated by the receiver. It is assumed that $s_k(t)$ spans D data bits as received from satellite k, where D typically ranges from 50 to 150 data bits (1-3 seconds of signal). Each signal $s_k(t)$ depends on the four parameters described above, as well as signal amplitudes $a_{k,n}$ and phases $\varphi_{k,n}$. The index n is a data bit index running from 1 to D. The amplitudes and phases are considered to be nuisance parameters involved in minimizing L, but are not of primary interest. They are allowed to vary from bit to bit, because in mobile operation a considerable amount of fluctuation is often experienced over multiple data bits from the same satellite, but with little variation over a single data bit.

During the n^{th} data bit, the satellite k signal replica $s_k(t)$ has the form

$$s_k(t) = a_{k,n} m_k(t) \exp\left[j\left(\omega_k t + \phi_{k,n} \right) \right] \qquad (2)$$

where $m_k(t)$ is the C/A code modulation normalized to unity magnitude and ω_k is the radian/sec frequency. After expanding expression (1) for L, it is noted that it has the form $L = R - J$, where R depends only on $r(t)$ and contains no parameters. Thus, minimization of L is equivalent to *maximization* of J. After partially maximizing J with respect to the amplitudes and phases, and renormalizing it to account

for constants appearing in the partial maximization process, we obtain

$$J = \sum_{k=1}^{N} \sum_{n=1}^{D} \left| \int_{I_{k,n}} r(t)\, m_k(t) \exp\left(-j\omega_k t\right) dt \right|^2 \qquad (3)$$

where $I_{k,n}$ is the time interval containing the n^{th} of the D data bits from satellite k.

Interpretation of Expression (3) and Advantages of the ML Estimator

Inspection of (3) reveals some mathematical operations understood by designers of GPS receivers. Each of the ND integrals is basically a correlation of the received baseband signal with the product of a replica PN code and a complex-valued cissoid. For each satellite, the inner summation simply sums the squared magnitudes of D correlations, each correlation occurring over a data bit. This summation is the familiar approach to obtain non-coherent processing gain over multiple data bits.

What is probably not familiar to most designers is the outer summation, which sums over all satellites. This operation achieves additional processing gain at low signal-to-noise ratios which is not exploited in typical GNSS receivers (even those which use VDFLL tracking loops). The result is improved position and velocity accuracy, as well as lowered tracking thresholds.

To gain an appreciation for the effect of the outer summation in the ML estimator, take a look at Figure 4. The top five panels show the values of the inner sum for each of 5 weak satellite signals as a function of ω_k, k = 1,2,3,4,5 (for simplicity, it is assumed that the code functions $m_k(t)$ are already maximizing J). In a conventional GPS receiver, the frequency at which the inner sum for the k^{th} satellite is maximum is a measurement of the frequency error for that satellite, the error being fed to the navigation processor. These points are shown as red dots. In the figure the true value of frequency error is zero for all five satellites. However, because the signals are so weak, there are outliers in the frequency measurements which will disruptively propagate through the navigation processor. In contrast, the ML estimator obtains processing gain by summing the functions in the top five panels before making the velocity estimate. This is the outer sum in expression (3). In the bottom panel, it is seen that maximization of J with respect to receiver velocity has no outliers, as shown by the green dot.

Details of the Maximization of J

The maximization of J in (3) constitutes the important processing in the new MLVTL design. The parameters involved in the maximization, which are implicit in the functions $m_k(t)$ and frequencies ω_k, are the receiver position $\mathbf{p}(T_0)$, residual GPS time bias $b(T_0)$, receiver velocity $\mathbf{v}(T_0)$, and residual GPS time rate bias $\dot{b}(T_0)$, all at time T_0. For our purposes the receiver position and velocity vectors are respectively in meters and meters/sec in a local coordinate system tangent to the Earth's surface, and residual GPS time bias and residual GPS time rate bias are respectively in seconds and seconds/sec. It is assumed that receiver velocity remains constant on the time interval $[T_0,T_0+T]$. The dependence of $m_k(t)$ and ω_k on these four parameters will now be made explicit.

Dependence of $m_k(t)$ on Parameters

The code waveform for satellite k has the close approximation

$$m_k(t) \cong M_k\left[t - b(T_0) - \tau_k(T_0) - (t - T_0)\,\tau_k'(T_0)\right] \qquad (4)$$

where $M_k(t_{GPS})$ is the code modulation as it is transmitted at the satellite expressed as a function of GPS time, $\tau_k(T_0)$ is the signal propagation delay in seconds from the satellite k to the receiver for the point on the signal that arrives at time T_0, and $\tau_k'(T_0)$ is the rate of propagation delay change at time T_0 in seconds/sec. The code modulation $M_k(t_{GPS})$ is a known function of GPS time which is slaved to the atomic clock in satellite k. Keeping in mind that t is the small-error estimate of GPS time provided by the receiver, the GPS time rate bias $\dot{b}(T_0)$ has been omitted from (4) because its effect is negligible over the time interval $[T_0,T_0+T]$.

The propagation delay $\tau_k(T_0)$ is closely approximated by

$$\tau_k(T_0) \cong \frac{\left|\mathbf{p}_k(T_0) - \mathbf{p}(T_0)\right|}{c + \mathbf{v}_k(T_0)\cdot\mathbf{u}_k(T_0)} \text{ sec} \qquad (5)$$

where at time T_0 the vectors $\mathbf{p}_k(T_0)$, $\mathbf{v}_k(T_0)$, and $\mathbf{u}_k(T_0)$ are respectively the position of satellite k, the velocity of satellite k, and the unit vector pointing from the most recent estimated receiver position to the position of satellite k. These vectors are computed from received satellite ephemeris data. The components of the position vectors are in meters and those of the velocity vector are in meters/sec. The speed of light is denoted by c, and is 2.99792458×10^8 meters/sec. Expression (5) takes into account the fact that the signal arriving at time T_0 was actually transmitted from the satellite position at an earlier time.

The rate $\tau_k'(T_0)$ of propagation delay change in (4) is closely approximated by

$$\tau_k'(T_0) \cong \frac{1}{c}\left\{\left[\mathbf{v}_k(T_0) - \mathbf{v}(T_0)\right]\cdot\mathbf{u}_k(T_0)\right\} \text{ sec/sec} \qquad (6)$$

In (4)-(6) the important parameters in the maximization of J are $\mathbf{p}(T_0)$, $b(T_0)$, and $\mathbf{v}(T_0)$. For given values of this triple, the code waveform $m_k(t)$ in expression (3) for J is computed by substituting $\mathbf{p}(T_0)$ into (5) to get $\tau_k(T_0)$, putting $\mathbf{v}(T_0)$ into (6) to get $\tau_k'(T_0)$, putting $b(T_0)$, $\tau_k(T_0)$, and $\tau_k'(T_0)$ into (4) to get $m_k(t)$, and then putting $m_k(t)$ into expression (3).

Dependence of ω_k on Parameters

The received baseband frequency ω_k from satellite k is closely approximated by

$$\omega_k = 2\pi \left\{ \frac{1}{\lambda} \left[\mathbf{v}\left(T_0\right) - \mathbf{v}_k\left(T_0\right) \right] \cdot \mathbf{u}_k\left(T_0\right) - \dot{b}\left(T_0\right) f_{\mathrm{L1}} \right\} \quad (7)$$

where $f_{\mathrm{L1}} = 1{,}575.42 \times 10^6$ Hz and $\lambda = 0.1902937$ meters are respectively the GPS L_1 frequency and wavelength. Here the important parameters in the maximization of J are $\mathbf{v}(T_0)$ and $\dot{b}\,(T_0)$, which determine ω_k.

The ML Estimate

The ML estimate of interest is the parameter vector $[\mathbf{p}(T_0),\ b(T_0),\ \mathbf{v}(T_0),\ \dot{b}\,(T_0)]$ which maximizes J in (3), with maximizing values denoted by hats:

$$\text{ML Estimate} = \left[\hat{\mathbf{p}}\left(T_0\right), \hat{b}\left(T_0\right), \hat{\mathbf{v}}\left(T_0\right), \hat{\dot{b}}\left(T_0\right) \right] \quad (8)$$

It should be noted that the terms in (3) are coupled by virtue of the four common parameters. Therefore, the terms cannot be independently maximized.

The calculations of the ML estimate require various values of the estimated GPS time t. This time is available from a corrected receiver clock. The method of providing the corrected clock, including initialization, will be discussed later when a tracking architecture is developed.

ML Estimator Block Diagram

A block diagram of the ML estimator is shown in Figure 5. A search over the parameters $\mathbf{p}(T_0)$, $b(T_0)$, $\mathbf{v}(T_0)$, and $\dot{b}\,(T_0)$ is required in order to generate the functions $m_k(t)$ and frequencies ω_k in expression (3) which maximize J. The extent of the search is determined mostly by the uncertainty in receiver position $\mathbf{p}(T_0)$ and velocity $\mathbf{v}(T_0)$. The set of values of the parameter vector $[\mathbf{p}(T_0),\ b(T_0),\ \mathbf{v}(T_0),\ \dot{b}\,(T_0)]$ used in the search is called the *search space*, which consists of four subspaces respectively called the *position, GPS time bias, velocity, and GPS time bias rate* search spaces. The search space and its four subspaces will be discussed more fully in the next section, which describes the new MLVTL tracking loop.

3. THE MLVTL TRACKING ARCHITECTURE

Initialization for Tracking

The standard method for initializing the four navigation parameters is to use well-known methods in the art to independently acquire the satellite signals, record ephemeris data, obtain bit synchronization, make independent pseudorange and frequency measurements, and use this data to obtain initial estimates of position, velocity, and GPS time, thus establishing the first search space center for tracking.

Tracking Method

To track receiver position, velocity, GPS time and time rate, a simple tracking method with good performance for ground-based mobile applications is to simply repeat the basic ML estimation process previously discussed over a sequence of *signal time intervals* $[T_0,T_0+T]$, $[T_1,T_1+T]$, $[T_2,T_2+T]$, … , $[T_m,T_m+T]$, … , where the time difference T_R between successive interval starting points is

$$T_R = T_{m+1} - T_m, \quad T \le T_R \quad (9)$$

In moving from one signal time interval to the next, the *search space center* is first updated according to the equations

$$\begin{aligned} \tilde{\mathbf{p}}\left(T_{m+1}\right) &= \hat{\mathbf{p}}\left(T_m\right) + \hat{\mathbf{v}}\left(T_m\right) T_R \\ \tilde{b}\left(T_{m+1}\right) &= 0 \\ \tilde{\mathbf{v}}\left(T_{m+1}\right) &= \hat{\mathbf{v}}\left(T_m\right) \\ \tilde{\dot{b}}\left(T_{m+1}\right) &= 0 \end{aligned} \quad (10)$$

where the tilde denotes the value at the search space center. The reason that the search space centers for GPS time bias and bias rate are set to zero is that GPS time t is being generated by a GPS time generation algorithm which uses filtered estimates of bias residuals to maintain GPS time very accurately from one signal time interval to the next. The filtering takes advantage of the inherent stability of the receiver clock to substantially improve the GPS time accuracy.

A method for generating GPS time from the receiver clock is shown in Figure 6. A register holds the increment Δt of GPS time t which occurs during each period T_c of the receiver clock. At each tick of the receiver clock, this increment is added to an accumulator holding GPS time t. Each iteration of the tracking loop (one every T_R seconds) provides a filtered estimate $\bar{b}$ of the residual GPS time bias and a filtered estimate $\bar{\dot{b}}$ of the residual bias rate which are used as corrections to GPS time. The Δt register is updated by setting it with $\bar{\dot{b}}$ and the GPS time t accumulator is updated by subtracting $\bar{b}$ from its contents.

Figure 7 is a high-level block diagram showing the basic elements of the tracking loop. Note that receiver acceleration is not taken into account in tracking, since large accelerations in ground-based mobile operation are never sustained. The loop design permits quick recovery from them when smaller accelerations invariably follow the large ones. As will be seen in Section 5, the MLVTL can perform quite well with moderate accelerations, even if they are sustained.

Description of Search Spaces

The search space for position (or velocity) is shown Figure 8. The space is two-dimensional, since it is assumed that altitude is known and movement is essentially horizontal. Prior to the search using the signal on the next interval $[T_{m+1},T_{m+1}+T]$, the space is centered at the updated position $\tilde{\mathbf{p}}(T_{m+1})$ [or velocity $\tilde{\mathbf{v}}(T_{m+1})$] in accordance with expression (10). The position resolutions Δx, Δy (or the velocity resolutions Δv_x, Δv_y) are chosen to obtain a good compromise between the number of points required in the search and the granularity permitting adequate accuracy.

The size of the search space for position depends on the maximum expected change in receiver velocity from estimate to estimate, the desired position resolution in the search space, and the amount of position constraint available from map aiding. A typical maximum expected velocity change over time T_R = 2 seconds might be 20 meters/sec, so if T_R = 2 seconds, a conservative search radius might be somewhat more than 40 meters. A typical position resolution might be 5 meters, which can be interpolated down to 1 meter or less.

The size of the search space for velocity depends on the maximum expected change in receiver velocity from estimate to estimate, the desired velocity resolution in the search space, and the amount of velocity constraint available from map aiding. As previously stated, a typical maximum expected velocity change over time T_R = 2 seconds might be 20 meters/sec, so if T_R = 2 seconds, the search radius might be somewhat more than 20 meters/sec. A typical velocity resolution might be 5 meters/sec, which can be interpolated down to 1 meter/sec or less.

The radii of the position and velocity search spaces can be made adaptive. If the maximizing position (or velocity) for J occurs on the boundary circle or if a definitive maximum is not found within it, the circle can be expanded on the fly. This is useful for emerging from severe signal outages which might last longer than the ML update interval T_R.

Because the receiver-generated GPS time is very stable and not subject to vehicle dynamics, the search spaces for $b(T_m)$ and $\dot{b}(T_m)$ can be extremely small, perhaps at most 3 points each with close spacing. Furthermore, the estimates of these parameters need not be performed every time the position and velocity estimates are updated, because their rate of variation is so slow.

Search Strategies

Several methods of conducting the search are possible. A brute force search assures finding a global maximum of J, and computation can be minimized by first performing a coarse search followed by a finer one. Alternatively, a hill-climbing method, which is essentially a discrete gradient approach, can be used to reduce search computation if conditions are such that a local maximum is guaranteed to be a global one.

Map Aiding

The position and velocity search spaces can be restricted to conform to a road when map aiding is available, as shown in Figure 8. It can be seen that map aiding can greatly reduce the size of the search space by making only certain positions and velocities admissible in the search. Map aiding also provides the advantage that the reduction in the degrees of freedom of these parameters reduces estimation errors.

Minimum Number of Satellites for Tracking

Without map aiding, a minimum of three satellites suffices for two-dimensional continuous tracking over any length of time, and two satellites are sufficient for "flywheel" operation over a time period which depends on the receiver oscillator stability. On the other hand, a major advantage of "in the loop" map aiding allowable by the MLVTL design is that only two satellites are sufficient for continuous tracking, and flywheel operation is possible with only one satellite (although in this case the tracking performance depends upon how closely the direction vector to the satellite lines up with the street direction).

4. TRACKING IMPLEMENTATION

Fortuitously, the nature of J is such that it can be maximized with very little error without potentially having to traverse all points in the full four-dimensional search space. Because the magnitude of J with respect to position is maintained over a relatively broad range (perhaps ± 100 meters or more), it is relatively insensitive to departure from the position search space center. Therefore, maximization can first be made with respect to velocity and GPS time bias rate. The maximizing velocity is then held constant while further maximizing with respect to position and GPS time bias. This procedure will require that the signal on time interval $[T_m,T_m+T]$ be captured in memory so that it can be accessed more than once. If desired, a final touchup of the maximizing velocity can be performed.

A tracking implementation which takes advantage of this reduced search space computation is shown in Figures 9 and 10, in which the iteration using the received signal on time interval $[T_m,T_m+T]$ is shown. Two computational steps are involved (for purposes of simplicity, the maximization of J with respect to GPS time bias and bias rate is omitted):

Step 1 (Figure 9): Maximize *J* With Respect to Velocity

Consider the signal from satellite *k*, embedded in the composite baseband signal. The signal is first frequency-corrected by its expected received frequency $\tilde{\omega}_k$ at the estimated velocity search space center $\tilde{\mathbf{v}}(T_m)$. Then the signal undergoes a sequence of 1-millisecond correlations with the expected received code waveform $\tilde{m}_k(t)$ at the estimated position search center $\tilde{\mathbf{p}}(T_m)$. There are 20 such correlations within each of *D* received data bits, and the correlations are time-aligned to fit within the data bit boundaries established by initial bit synchronization and tracking. Each group of 20 correlation outputs is fed to an FFT. The squared magnitudes of corresponding FFT bin outputs are accumulated over the time interval *T*, and the accumulated results are stored. The stored accumulated values for satellite *k* constitute a set of possible *frequency residuals*, and span a frequency range from −500 to 500 Hz. A search over trial *velocity navigation residual values* $\Delta\mathbf{v}(T_m)$ is then conducted. These trial values are departures in velocity from the velocity search space center $\tilde{\mathbf{v}}(T_m)$, and each value is mapped into a selected corresponding value of frequency residual for satellite *k*.

The selected frequency residuals from all satellites are summed to produce a value for *J*. The value of $\Delta\mathbf{v}(T_m)$ which maximizes *J* is denoted by $\widehat{\Delta\mathbf{v}}(T_m)$. The estimate of receiver velocity $\hat{\mathbf{v}}(T_m)$ is then computed as

$$\hat{\mathbf{v}}(T_m) = \tilde{\mathbf{v}}(T_m) + \widehat{\Delta\mathbf{v}}(T_m) \qquad (11)$$

Step 2 (Figure 10): Maximize *J* With Respect to Position

Again consider the signal from satellite *k* within the composite baseband signal. The signal is now frequency-corrected by the frequency $\hat{\omega}_k$ determined by the velocity estimate $\hat{\mathbf{v}}(T_m)$ that was obtained in Step 1, resulting in a residual frequency which is small enough to permit 20-millisecond correlations to be performed. The signal then undergoes a sequence of *D* 20-millisecond multi-delay correlations with the expected received code waveform $\tilde{m}_k(t)$ at the estimated position search center $\tilde{\mathbf{p}}(T_m)$. The number of correlation delays produced in each multi-delay correlation depends on the size of the position search space and the desired position resolution within the space, but will usually span only a fraction of a microsecond because a single C/A code correlation function peak spans the relatively large range of from about −1 to 1 microseconds (approximately −300 to 300 meters). For each correlation delay, the squared magnitudes of the *D* correlation outputs are accumulated over the time duration *T* and stored. The stored accumulated values for satellite *k* constitute a set of possible *delay residuals*, and span a range which normally would be smaller than from −1 to 1 microseconds (approximately −300 to 300 meters). A search over trial *position navigation residual values* $\Delta\mathbf{p}(T_m)$ is then conducted. The trial values are departures in position from the position search space center $\tilde{\mathbf{p}}(T_m)$, and each value is mapped into a selected corresponding value of delay residual for satellite *k*.

The selected delay residuals from all satellites are summed to produce a value for *J*. The value of $\Delta\mathbf{p}(T_m)$ which maximizes *J* is denoted by $\widehat{\Delta\mathbf{p}}(T_m)$. The estimate $\hat{\mathbf{p}}(T_m)$ of receiver position is then computed as

$$\hat{\mathbf{p}}(T_m) = \tilde{\mathbf{p}}(T_m) + \widehat{\Delta\mathbf{p}}(T_m) \qquad (12)$$

This completes Step 2, and now the receiver is ready to perform Step 1 and Step 2 on the signal from the next signal time interval $[T_{m+1}, T_{m+1}+T]$.

5. EXAMPLES OF PERFORMANCE

Tracking performance of the MLVTL was measured using MATLAB simulations.

Figure 11 is an example of the constant-velocity positioning performance gain achievable with the MLVTL using 3 satellites, as a function of received signal power, assumed the same for all satellites.

The red curve is a plot of RMS radial positioning error for a recursive least-squares estimator using a typical update interval of 100 milliseconds (5 bits of navigation data). This estimator is similar in performance to the Kalman filter commonly used with conventional VDFLL tracking loops.

The blue curve shows the VDFLL error achievable by a nonrecursive least-squares estimator repeatedly operating on 2-second segments of signal data. The data fed to this estimator consists of a pseudorange residual measurement and pseudorange rate residual measurement for each satellite.

The green curve shows the error using the MLVTL estimator described in this paper. The signal dwell time is 2 seconds. Not only is the RMS radial position error significantly smaller than the other two estimators at low received signal power, but the tracking threshold is also improved by approximately 2 dB.

Figure 12 is similar to Figure 11, but it compares the velocity errors of three estimators at constant velocity.

Figure 13 shows the dependence of position and velocity MLVTL tracking errors on signal power level at constant velocity when 6 satellites are available. Not only are the errors significantly smaller than with 3 satellites, but the tracking threshold is reduced from −161 dBm to −164 dBm.

Figure 14 illustrates the improvement in MLVTL position tracking performance at constant velocity when map aiding is used with 3 available satellites. In comparison to the

performance without map aiding shown in Figure 11, the tracking threshold is improved by approximately 3 dB and the positioning error is significantly smaller.

Figure 15 is a convincing illustration of the tracking robustness of the MLVTL in the presence of severe but unsustained acceleration. The receiver is tracking 3 satellites, each with a −160 dBm signal (14 dB-Hz C/N_0). The receiver moves steadily eastward at 30 meters/sec (67 mph), makes an instantaneous left turn, and continues northward at the same speed. Neither code nor carrier tracking is lost, despite the fact that the turn is so violent that it is physically impossible! It is believed that no receiver in production today could maintain lock with such a maneuver, especially with the extremely weak signals in this scenario.

The MLVTL can also track sustained but moderate accelerations quite well, as is shown in Figure 16. Here the vehicle is traversing a circular arc of radius 200 meters at a speed of 30 meters/sec (67 mph). The radial acceleration is 0.46 g. The receiver is tracking 3 satellites, each with a −155 dBm signal (19 dB-Hz C/N_0). The tracking threshold is somewhat elevated above the −160 dBm level because the sustained radial acceleration causes some frequency smearing during the noncoherent averaging used in the ML estimation process.

6. SUMMARY

The work presented in this paper has resulted in a new MLVTL architecture with performance improvement over conventional VDFLL tracking methods. The new architecture takes advantage of the non-sustainability of large accelerations in ground-based mobile platforms, and has very low tracking thresholds made possible by ML estimation using a large amount of noncoherent processing. Tracking sensitivity is augmented in the ML estimation process by using joint signal residuals instead of individual measurement residuals from discriminators. The result is a lowering of tracking thresholds by 2 dB or more, depending on the number of satellites used. The limited tracking range of typical discriminators is avoided, and the tracking range can be made adaptive to enable seamless recovery from complete dropouts of all signals.

The use of ML estimation eliminates the disadvantages of recursive least-squares or Kalman filters in the tracking loops, such as the need for covariance matrix control or potential numerical instability problems.

The MLVTL also naturally permits map aiding to be used directly in the tracking loop as opposed to conventional techniques of making adjustments to the unaided post-navigation solution. In-the-loop map aiding has the advantage that it reduces position/velocity errors, lowers tracking thresholds, reduces the search space size, and reduces the minimum number of satellites required.

Three satellites are sufficient for continuous unaided operation, and 2 satellites are sufficient for flywheel operation using the stability of the receiver clock. With map aiding, 2 satellites are sufficient for continuous operation, and 1 satellite is enough for flywheel operation.

7. REFERENCES

1. M. Lashley and D. Bevly, "GNSS Solutions: What are Vector Tracking Loops, and What Are Their Benefits and Drawbacks?" *Inside GNSS* Magazine, Gibbons Media & Research LLC, Vol. 5/Number 5, May/June 2009, pp. 16-21.

2. D. Benson, "Interference Benefits of a Vector Delay Lock Loop (VDLL) GPS Receiver," Proceedings of the 63rd Annual Meeting of the Institute of Navigation, Cambridge, Massachusetts, Institute of Navigation, April 2007.

3. J. Spilker, "Fundamentals of Signal Tracking Theory," *Global Positioning System: Theory and Applications, Vol. 1, Progress in Astronautics and Aeronautics*, Vol. 163, AIAA, Washington, D.C., 1996.

4. M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, Wiley & Sons, New York, 2001, pp. 71-76.

8. FIGURES

Figures are shown on the following pages.

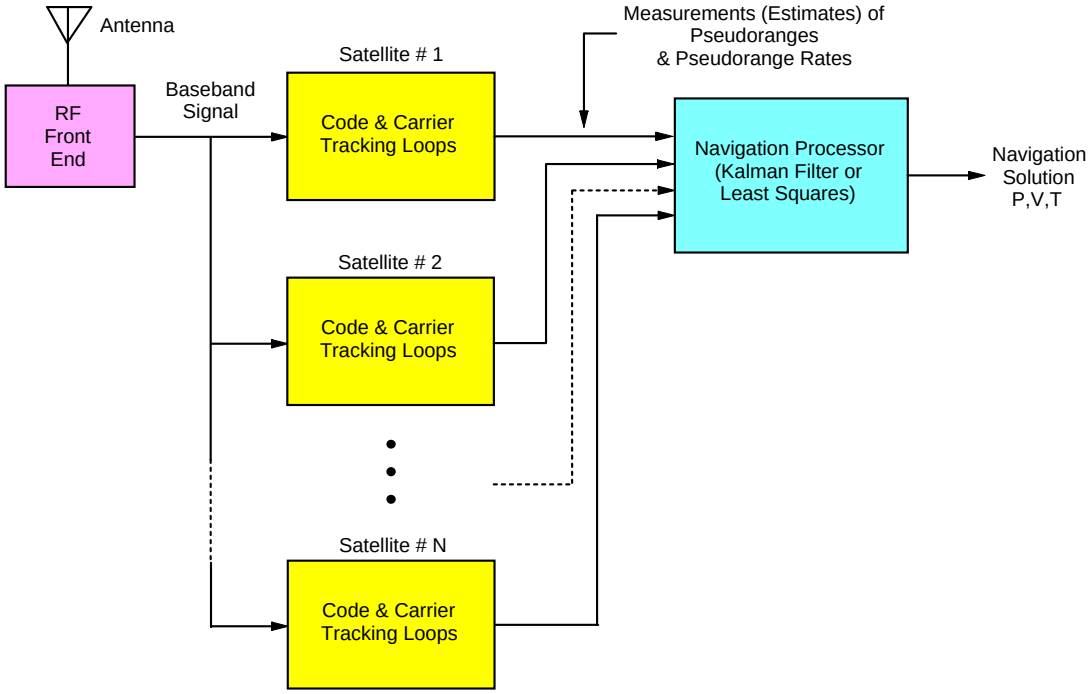

Figure 1. Conventional Scalar GNSS Signal Tracking Architecture

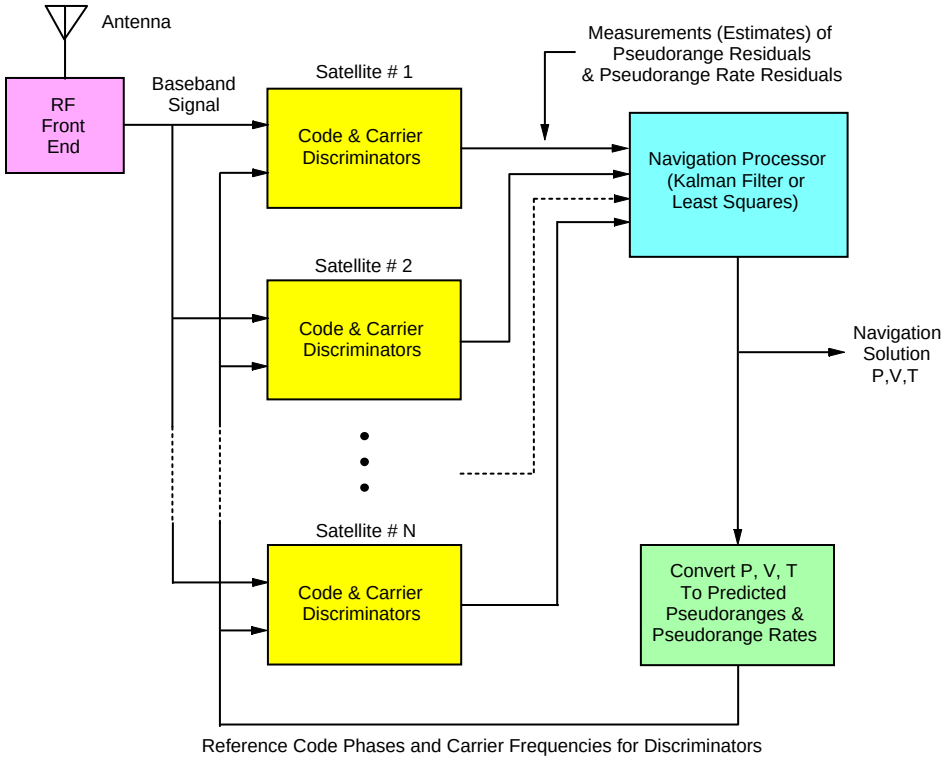

Figure 2. Typical Vector Tracking Loop (VFDLL)

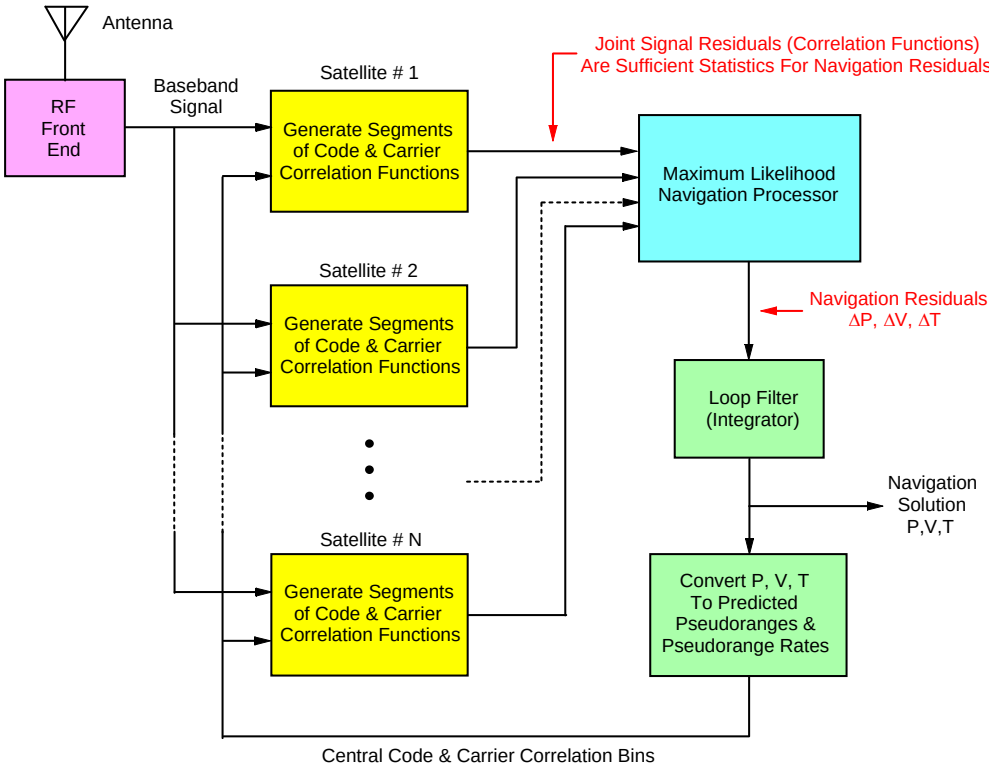

Figure 3. New ML Vector Tracking Loop (MLVTL)

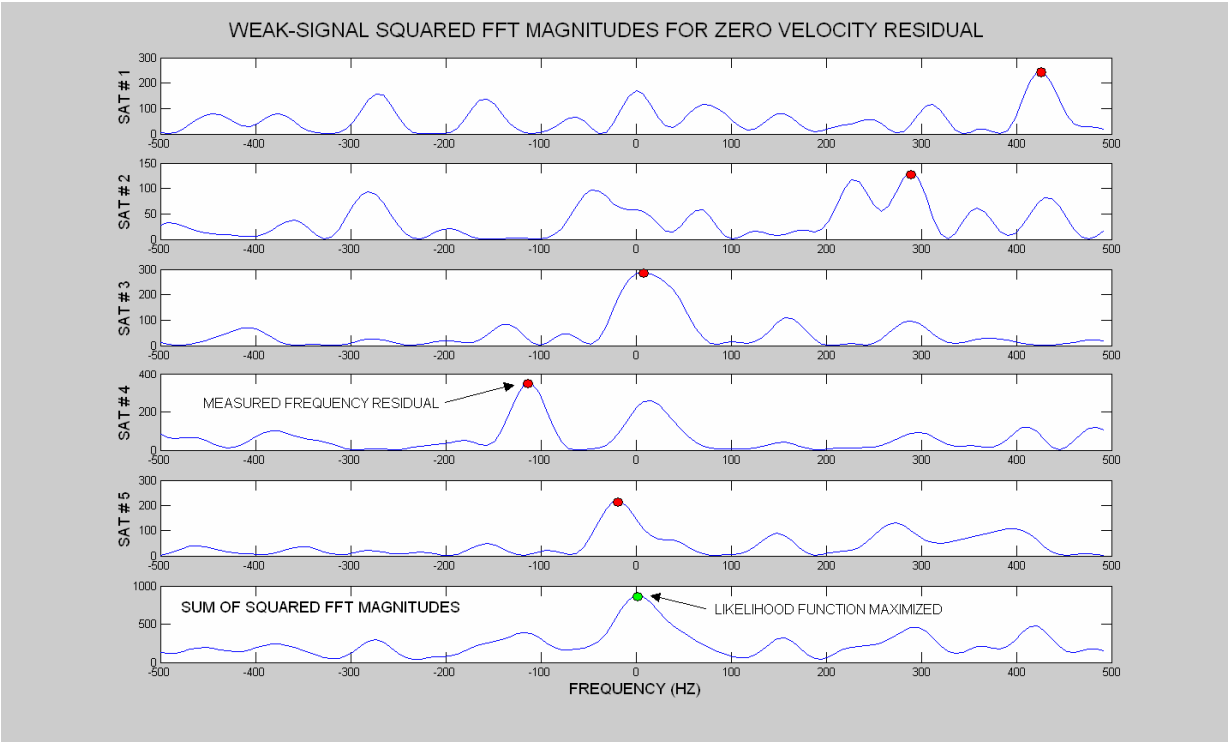

Figure 4. Frequency Residual Measurements Versus Noncoherent ML Combining

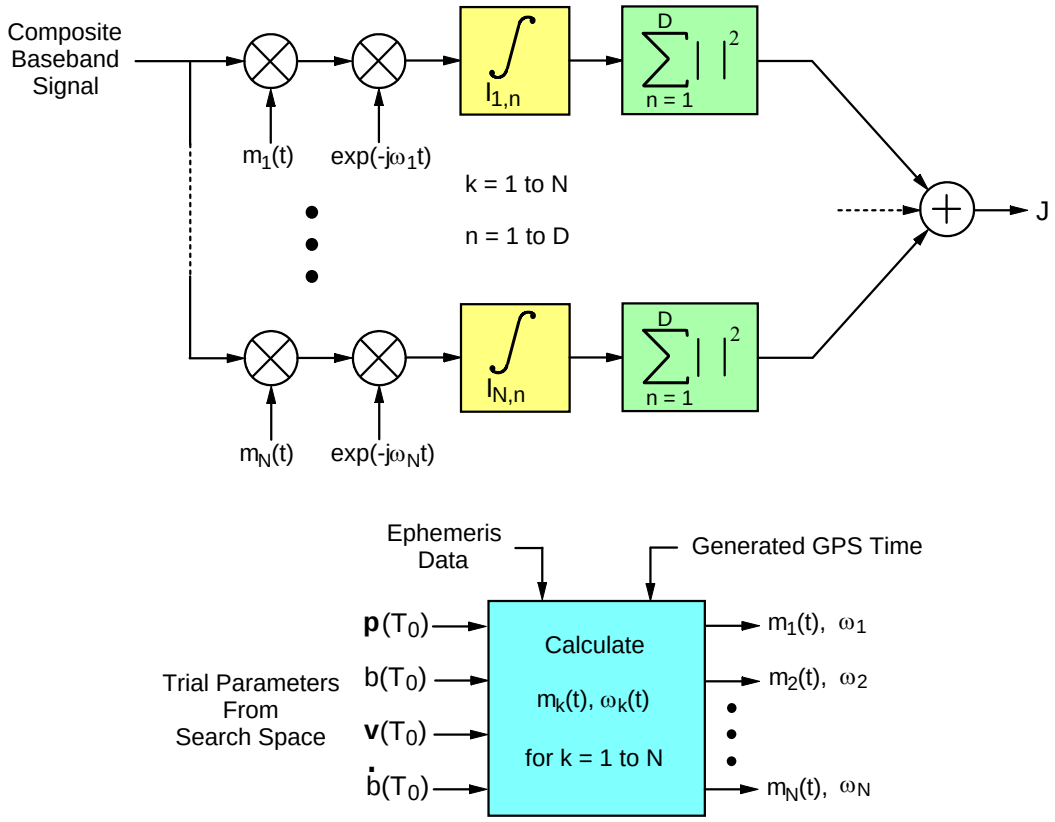

Figure 5. Block Diagram of the ML Estimator

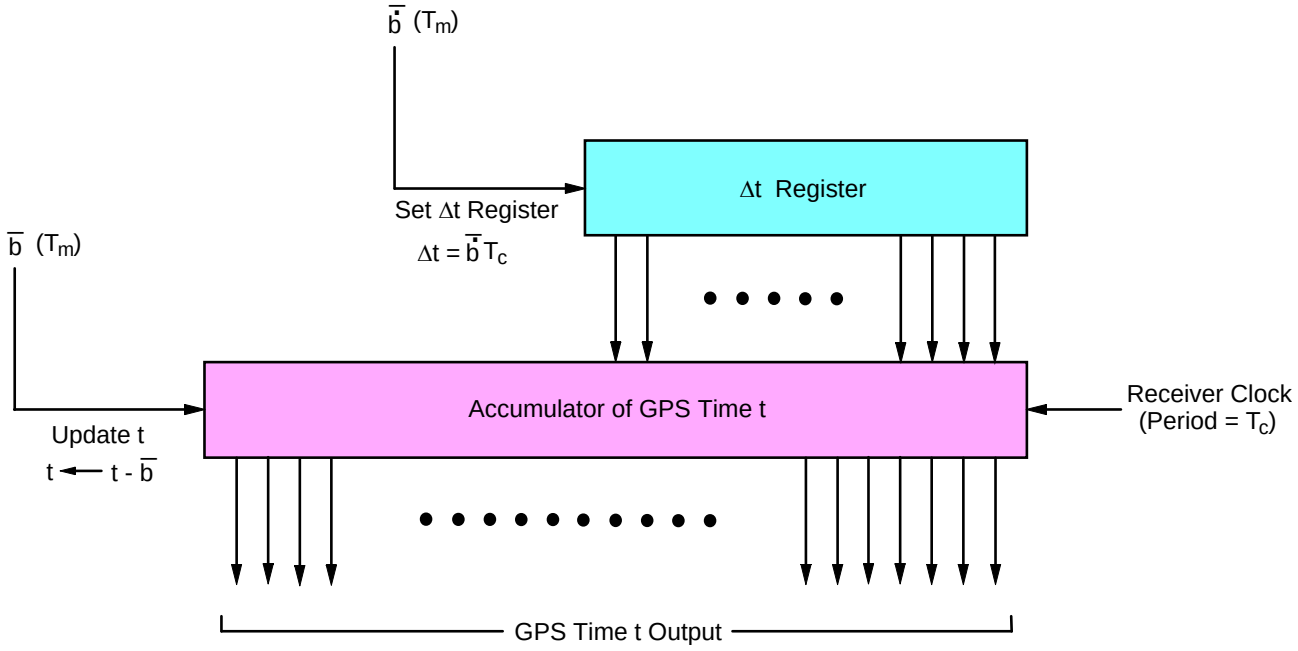

Figure 6. Method of Generating GPS Time

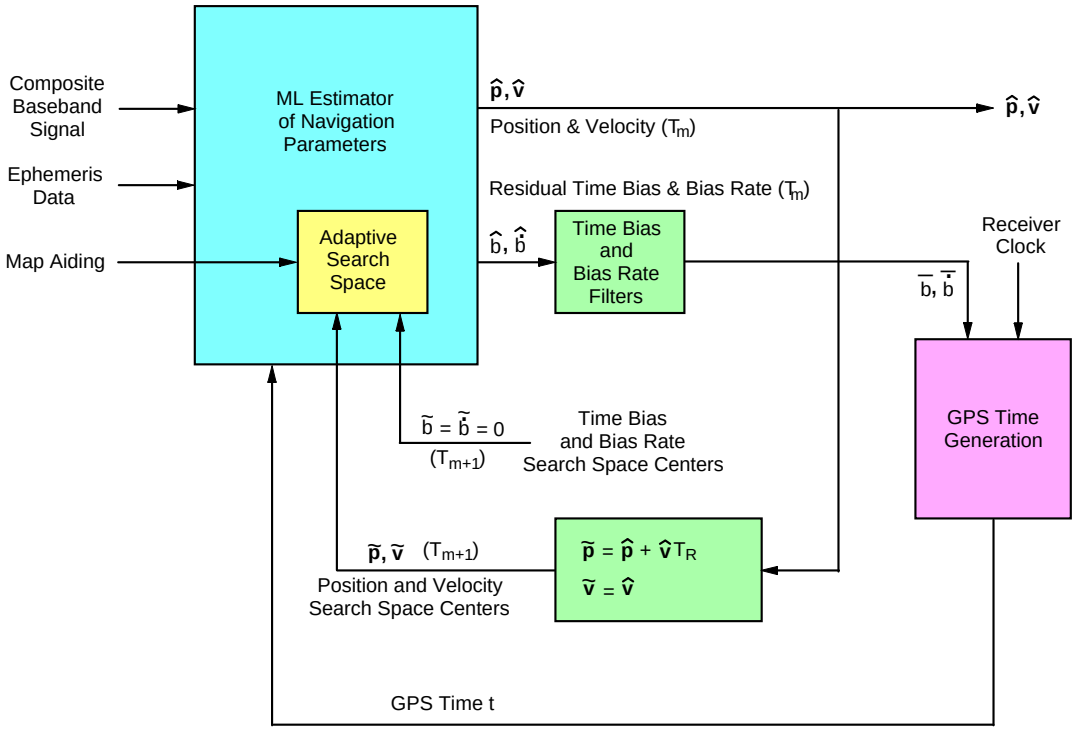

Figure 7. Basic Elements of the MLVTL

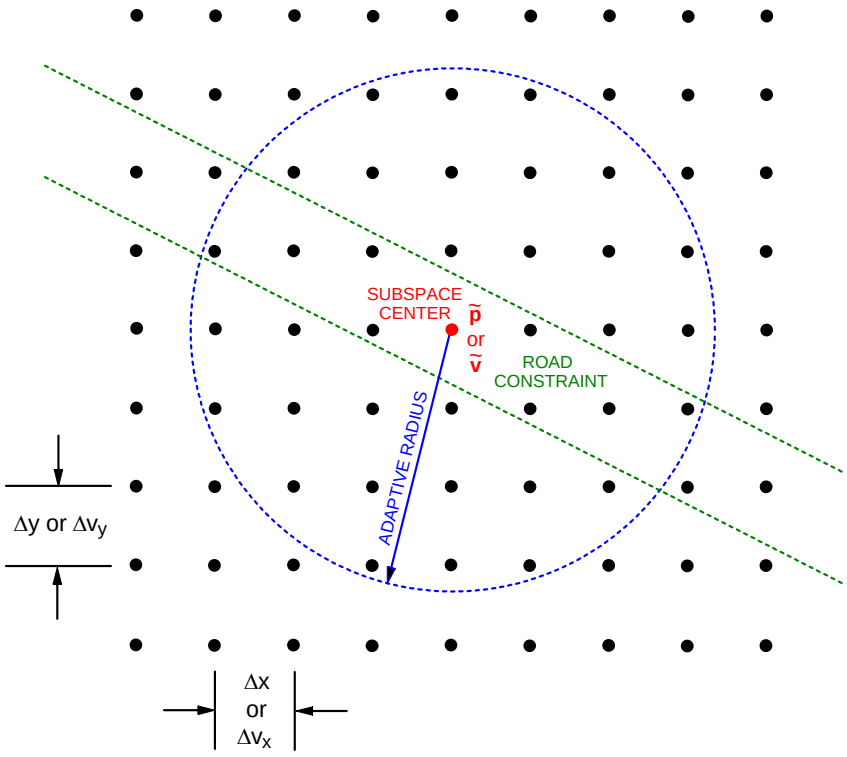

Figure 8. Position (or Velocity) Search Space

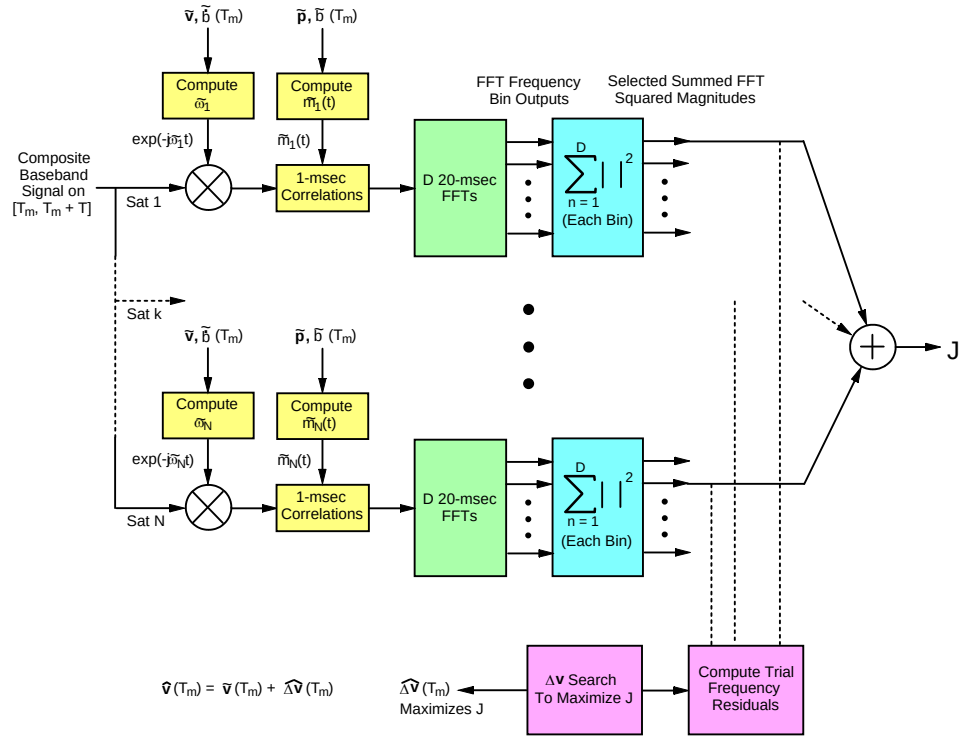

Figure 9. Step 1 of Tracking Implementation (Maximize *J* with Respect to Velocity)

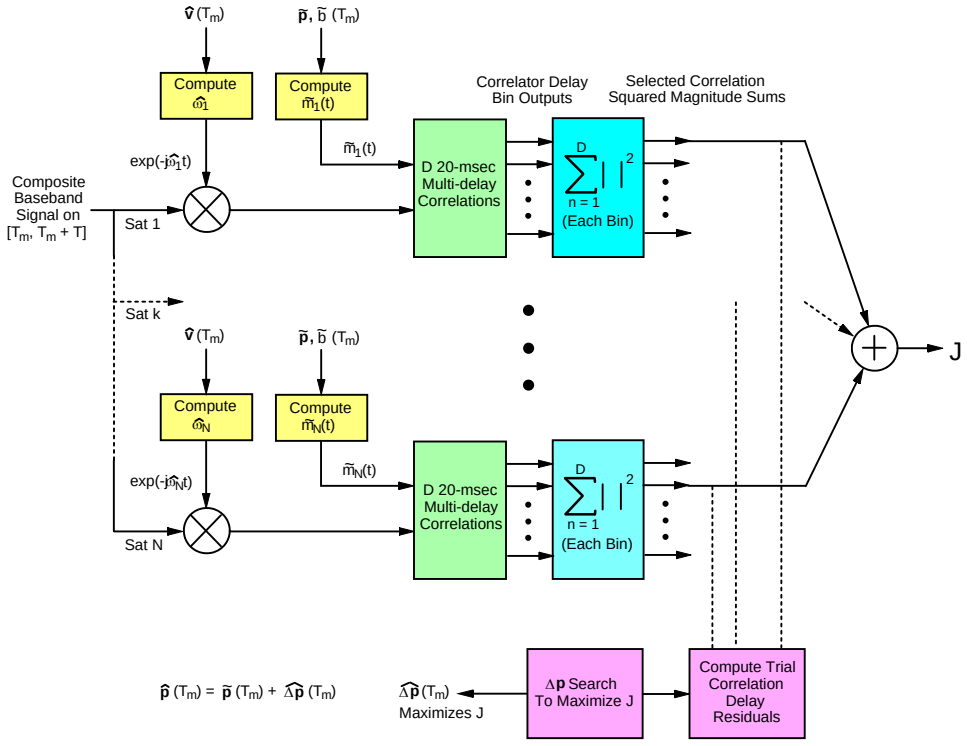

Figure 10. Step 2 of Tracking Implementation (Maximize *J* with Respect to Position)

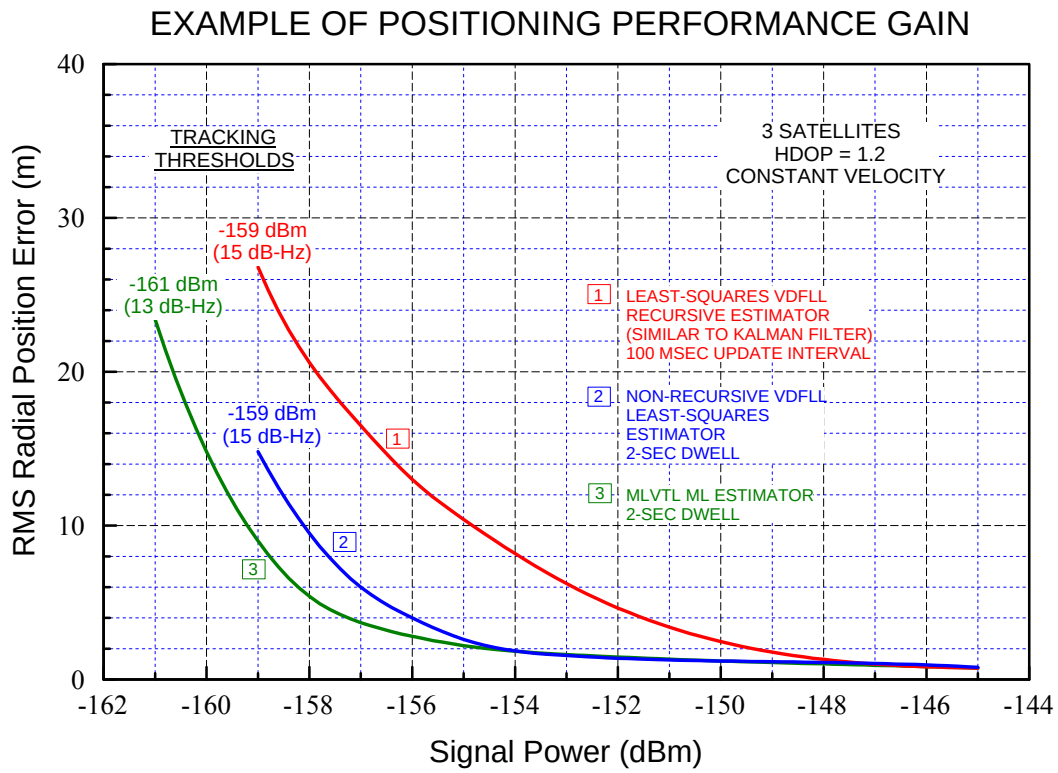

Figure 11. Constant-Velocity Position Tracking Comparisons

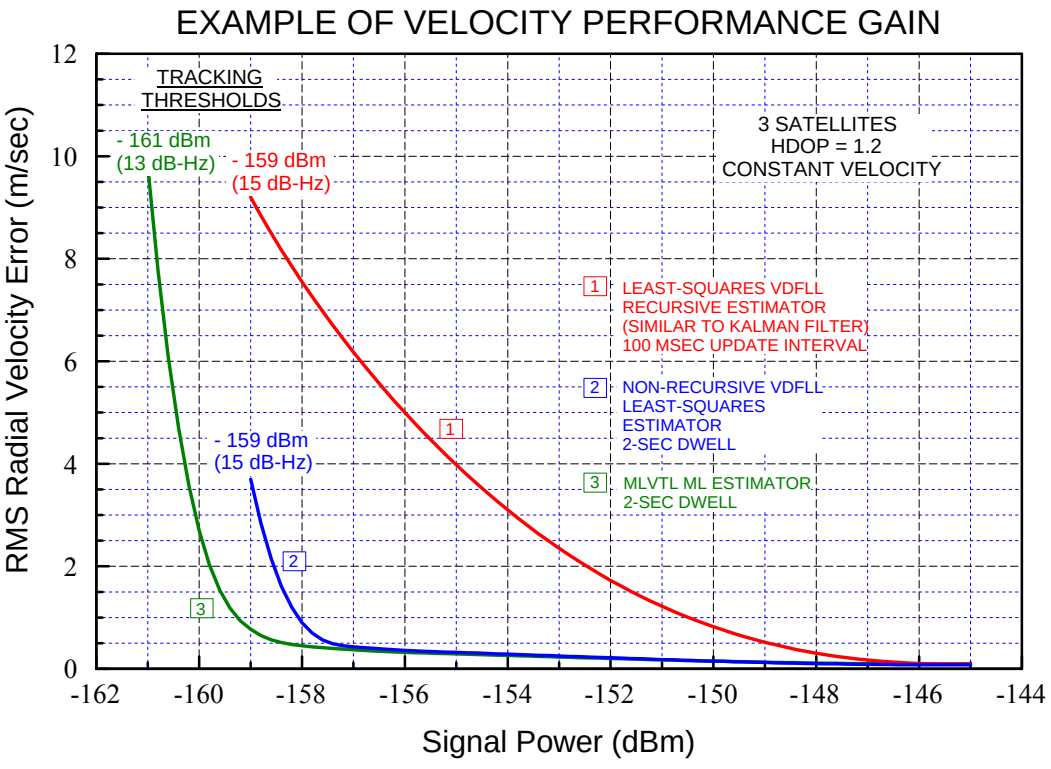

Figure 12. Constant-Velocity Velocity Tracking Comparisons

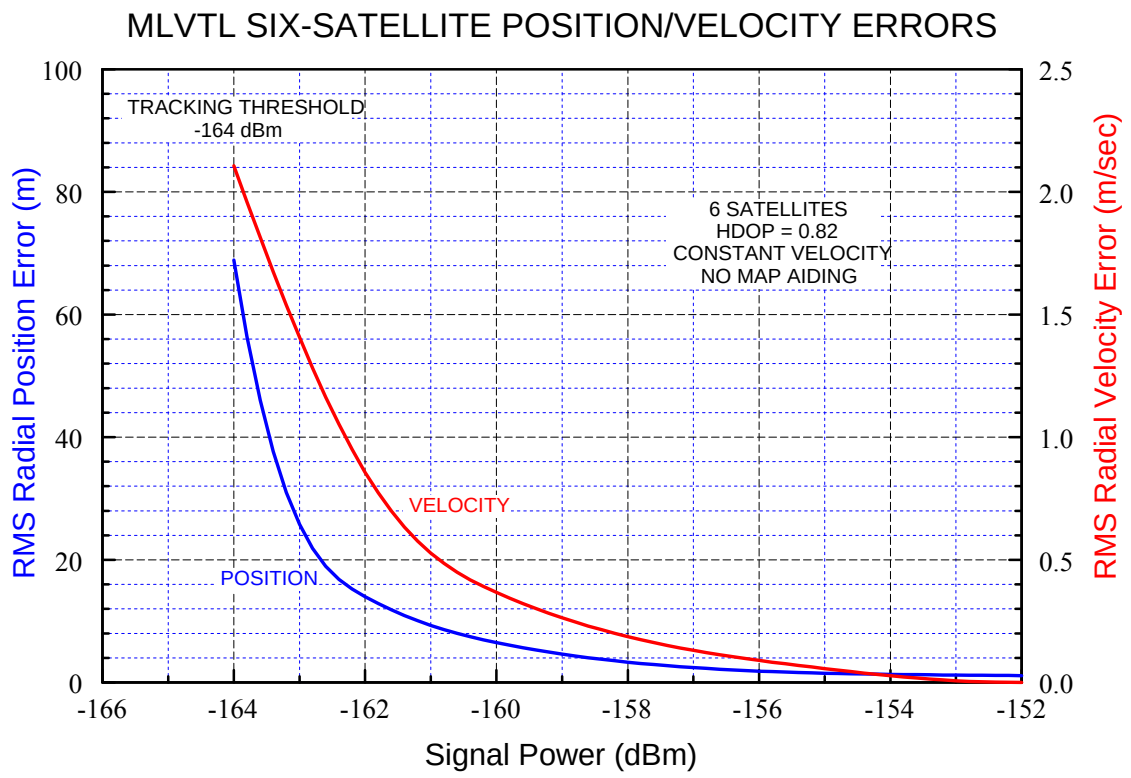

Figure 13. Six-Satellite Constant-Velocity Position and Velocity Performance

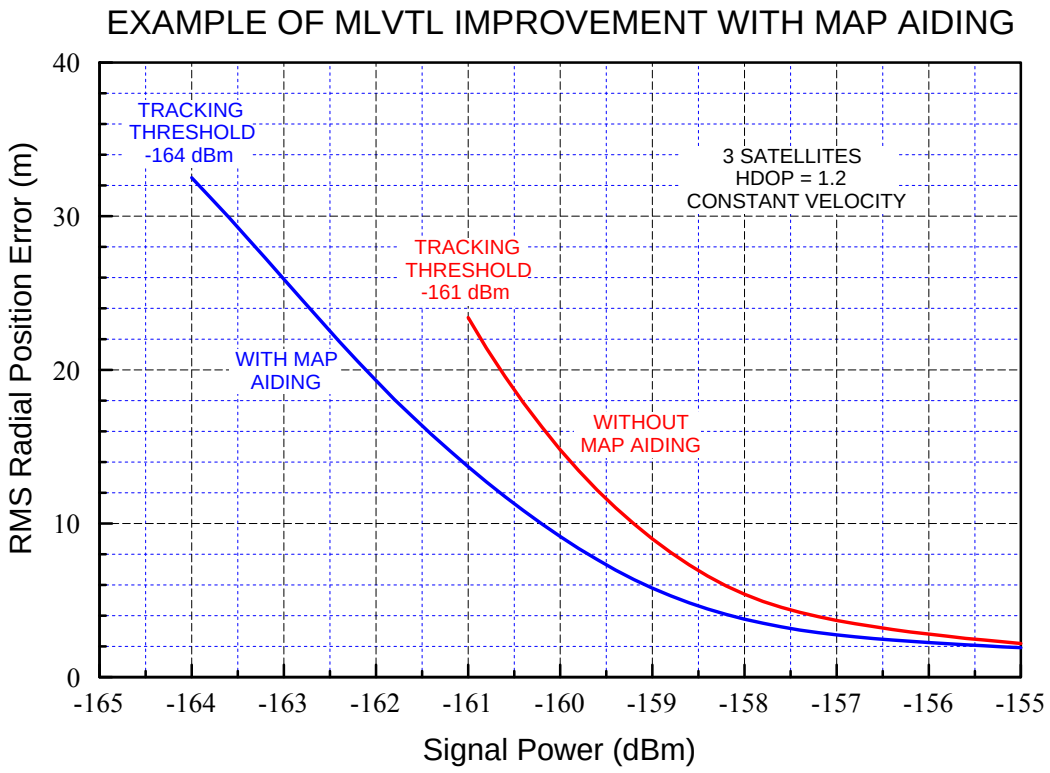

Figure 14. Position Tracking Performance Improvement with Map Aiding

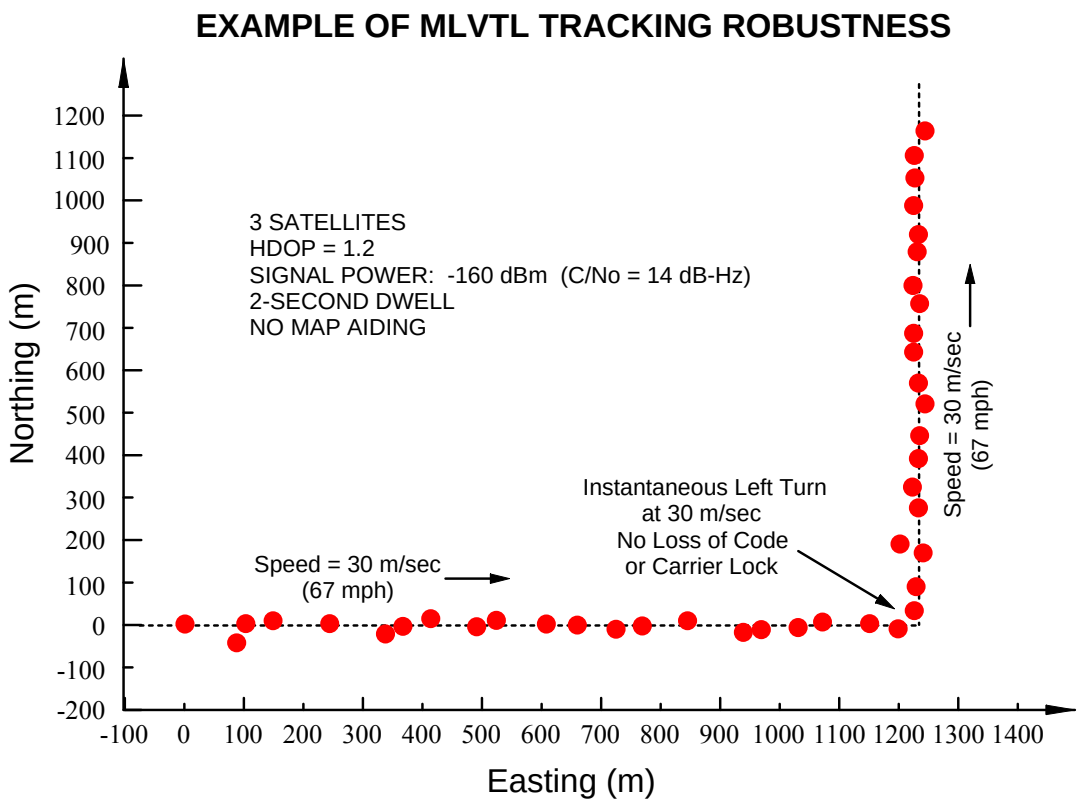

Figure 15. Weak-Signal Tracking Robustness with Large Momentary Accelerations

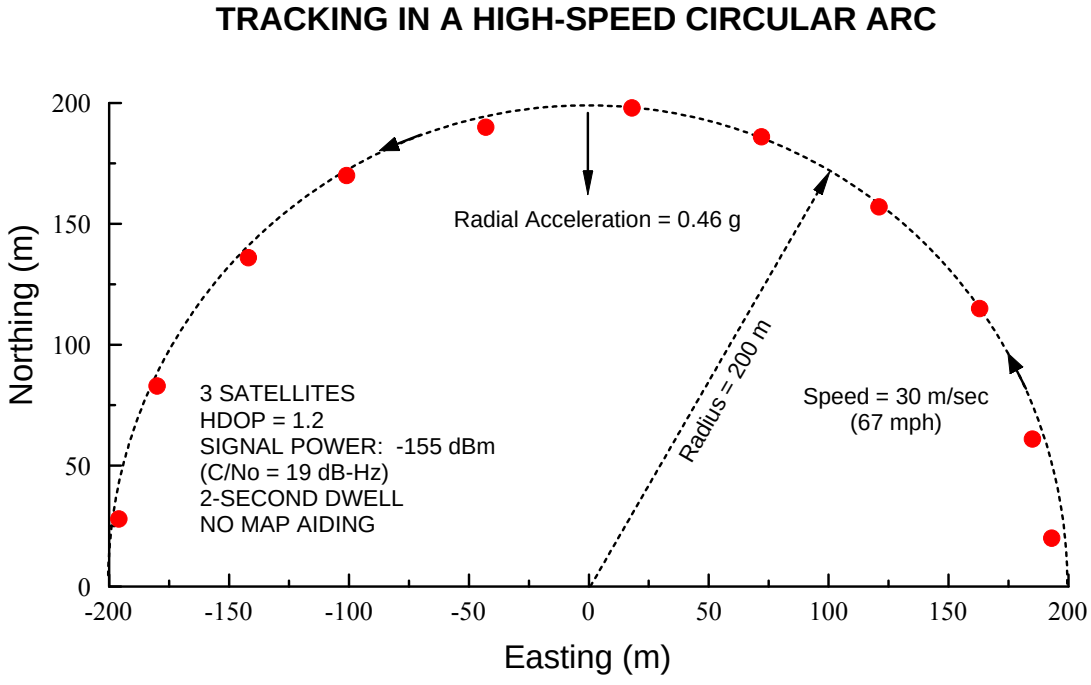

Figure 16. High-Speed Tracking with Sustained Radial Acceleration