Analysis of Discriminator Based Vector Tracking Algorithms

Matthew Lashley, *Auburn University*
David M. Bevly, *Auburn University*

BIOGRAPHY

Matthew Lashley is a research assistant in the GPS and Vehicle Dynamics Lab. He received his M.S. in EE from Auburn University in 2006 and his B.S. in E.E. from Auburn University in 2004. He is currently pursuing a PhD at Auburn University.

David M. Bevly is an assistant professor at Auburn University and head of the GPS and Vehicle Dynamics Lab. He received his PhD in Mechanical Engineering from Stanford University in 2001, his M.S. in M.E. from Massachusetts Institute of Technology in 1997, and his B.S. in M.E. from Texas A&M University in 1995.

ABSTRACT

In this paper, a variant of the vector delay lock loop (VDLL) tracking algorithm for tracking the L_1 civilian GPS signal is introduced. The algorithm functions on the principle originally described in [1]. The VDLL architecture uses a single Extended Kalman Filter to predict the satellite PRN code phase and track the user's position, velocity, and clock states. Additionally, a series of separate Kalman filters is used by each channel to track the satellite carrier signal. The advantages of the vector/Kalman filter based architecture over traditional methods are explained. The ability of the vector tracking algorithms to rapidly reacquire blocked signals is analyzed using data produced by a Spirent GPS simulator. The vector tracking algorithms performance is also compared to traditional methods using tracking loops. The VDLL is shown to outperform the traditional methods.

INTRODUCTION/BACKGROUND

Tracking loops are used by traditional receivers to track the signals broadcast by the GPS satellites. Typically, a Costas loop is used to track the carrier portion of the signal and a Delay Lock Loop (DLL) is used to track the Pseudo-Random Noise (PRN) sequence. The tracking loops are usually initialized with information from an acquisition program. Tracking loops operate by generating replicas of the received signals. The received and locally generated signals are multiplied together and the product is summed over an integration interval. The accumulated values are used to generate error signals which are proportional to the phase error between the received and replica signals. Tracking loops attempt to drive the phase error signals to zero, thereby maintaining phase-lock between the received and replica signals. This is accomplished by passing the phase error signals through a loop filter. The output of the filter is used to control the frequency of the replica signal.

Tracking loops operate well in environments with high carrier power-to-noise density ratio (C/No) levels and low user dynamics. However, they have several inherent flaws. First, tracking loops use loop filters with fixed gains and bandwidths. This means that all the phase error signals are equally weighted. Ideally, phase error measurements made during periods of high C/No levels should be weighted more heavily than those made during periods of low C/No levels. During times of satellite blockage, the loops are in a state of random walk. Second, the tracking loops in the different channels of the receiver operate independently of each other and don't exploit all the user's knowledge. Information about the satellite constellation and user position can be used to predict the received signals.

The Vector Delay Lock Loop (VDLL) combines the tracking of the different satellite PRN signals into a single algorithm, as introduced in [1]. The VDLL operates on the principle that the phases of the received PRN sequences are based on the user's position. So, given the satellite ephemeris, the user's position, and the user's clock bias, the PRN sequences from each satellite can be predicted. The VDLL uses estimates of the user's position to generate the replica PRN sequences. The error signals generated after each integration interval are then used to estimate the user's position and clock bias. An Extended Kalman Filter (EKF) is used to track the user's navigation states (position, velocity, and clock bias) and process the error signals generated after every integration interval, as described in [2]. In a VDLL, the tracking of the carrier signals is still handled independently in each channel.

The VDLL has several potential advantages over the standard DLL. First, the EKF can weight measurements of the code phase error. Therefore, provided the noise statistics of the individual measurements, the EKF can optimally estimate the user's states. Second, tracking weak signals can be facilitated by the tracking of stronger signals. This is due to the replica PRN sequences being controlled by the EKF's estimate of the user's position. Third, the VDLL has the potential to rapidly reacquire signals after a satellite blockage.

VDLL ARCHITECTURE

The VDLL developed in this paper uses a single EKF to track the PRN sequences received from the available satellites. The task of tracking the carrier portion of each received signal is handled independently in each channel. A Kalman filter based tracking algorithm is used instead of a Costas loop for the carrier signal. This allows the carrier tracking algorithm to weight measurements according to the noise present and ignore measurements made during periods of satellite blockage. Figure 1 shows a graphical representation of the VDLL architecture.

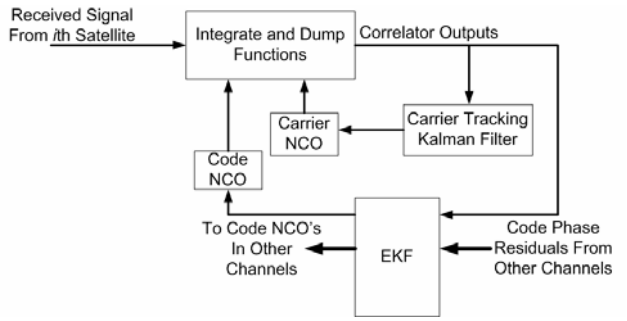

Figure 1: Vector Delay Lock Loop Architecture

The states of the individual carrier tracking Kalman filters are used to initialize the carrier Numerically Controlled Oscillators (NCO) in each channel. The carrier NCOs generate in-phase and quadrature replicas of the received carrier. The states of the single EKF are used to control the PRN code NCOs. The code NCOs produce early, prompt, and late replicas of the received PRN codes. At the end of each integrate and dump interval, the following six correlator outputs are produced in each channel:

IE = In-phase Early
IP = In-phase Prompt
IL = In-phase Late
QE = Quadrature Early
QP = Quadrature Prompt
QL = Quadrature Late

The expected values of the correlator outputs are derived in [3]. The correlator outputs for a single channel are used by that channel's carrier tracking Kalman filter. The correlator outputs for all channels are then used by the EKF.

The carrier tracking Kalman filter tracks the errors between the replica and received carrier signals [4]. The states and discrete time state transition matrix of the carrier tracking Kalman filter are shown in Equation (1).

$$\begin{bmatrix} \delta\theta_{c,k+1} \\ \delta f_{c,k+1} \\ \delta \dot{f}_{c,k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta T & 0 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta\theta_{c,k} \\ \delta f_{c,k} \\ \delta \dot{f}_{c,k} \end{bmatrix} \quad \textbf{(1)}$$

In the above equation, $\delta\theta_c$ is the phase error between the replica and received carrier, δf_c is the frequency error between the replica and received carrier, and $\delta \dot{f}_c$ is the error in the estimated rate of change of the carrier frequency. The rate of change in Doppler frequency of the carrier can be predicted from the user position, user velocity, and satellite ephemeris, or it can be estimated from phase and frequency measurements. The variable ΔT is the period of the integration (integrate and dump) interval.

Measurements for the carrier tracking Kalman filter are generated by using nonlinear discriminator functions. At the end of each integrate and dump operation, a two-quadrant arctangent discriminator is used to estimate the phase error between the replica and received carrier. The arctangent discriminator uses the in-phase and quadrature prompt outputs, as shown in Equation (2).

$$\theta_{error} \approx \arctan\left(\frac{QP}{IP}\right) \quad (2)$$

An estimate of the frequency error between the replica and received carrier signals is generated using the formulas shown in (3).

$$\begin{aligned} dot &= IP_{t1} \cdot IP_{t2} + QP_{t1} \cdot QP_{t2} \\ cross &= IP_{t1} \cdot QP_{t2} + IP_{t2} \cdot QP_{t1} \\ f_{error} &\approx \frac{\arctan\left(\frac{cross}{dot}\right)}{360(t_2 - t_1)} \end{aligned} \quad (3)$$

In the above formulas, the subscript *t1* and *t2* denote the accumulated values over the first and second half of the integrate and dump period, respectively. Since the carrier

tracking Kalman filter estimates the error in each state, the observation equations are as shown in Equation (4).

$$y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \delta\theta_{c,k} \\ \delta f_{c,k} \\ \delta \dot{f}_{c,k} \end{bmatrix} \quad (4)$$

The states and discrete time state transition matrix of the EKF used to track the PRN code phases are shown in Equation (5).

$$\begin{bmatrix} \delta x_{k+1} \\ \delta y_{k+1} \\ \delta z_{k+1} \\ \delta \dot{x}_{k+1} \\ \delta \dot{y}_{k+1} \\ \delta \dot{z}_{k+1} \\ \delta t_{k+1} \\ \delta \dot{t}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \Delta T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x_k \\ \delta y_k \\ \delta z_k \\ \delta \dot{x}_k \\ \delta \dot{y}_k \\ \delta \dot{z}_k \\ \delta t_k \\ \delta \dot{t}_k \end{bmatrix} \quad (5)$$

The states δx, δy, and δz correspond to errors in the estimated x, y and z coordinates of the user in the Earth Centered-Earth Fixed (ECEF) reference frame, respectively. Similarly, the states $\delta \dot{x}$, $\delta \dot{y}$, and $\delta \dot{z}$ correspond to errors in the user's estimated velocity in the x, y and z directions, respectively. The states δt and $\delta \dot{t}$ are the errors in the estimated user clock bias and drift, respectively. In Equation (5), the velocity states are modeled as constants. This is an incorrect assumption in most cases and can be remedied in one of two ways. One approach is to tune the process noise covariance matrix entries for the velocity states to accommodate the expected range of velocities for a specific user. A second approach is to integrate data from an Inertial Measurement Unit (IMU) as input to the velocity states

The correlator outputs from each channel are used by a normalized early-minus-late discriminator to produce measurements of the code phase error between the received and replica PRN codes, as shown in (6).

$$\varepsilon \approx \frac{\Sigma\sqrt{IE^2 + QE^2} - \Sigma\sqrt{IL^2 + QL^2}}{\Sigma\sqrt{IE^2 + QE^2} + \Sigma\sqrt{IL^2 + QL^2}} \quad (6)$$

The predicted code phases are based on the user's estimated position, clock bias, and the positions of the satellites. Predicting the code phases corresponds to predicting the pseudoranges to each satellite. The code phase errors can be considered as the difference between the predicted and actual pseudoranges, as shown in equation (7).

$$\begin{aligned} \varepsilon_i &= \rho_i - \hat{\rho}_i \\ \hat{\rho}_i &= \sqrt{(\hat{x}_u - x_i)^2 + (\hat{y}_u - y_i)^2 + (\hat{z}_u - z_i)^2} + c\hat{t}_u \end{aligned} \quad (7)$$

In equation (7), the variables $\hat{x}_u$, $\hat{y}_u$, $\hat{z}_u$, and $\hat{t}_u$ are the user's estimated position and clock bias. The variables x_i, y_i, and z_i are the position coordinates of the *i*th satellite. Using the relations in (7), the familiar linearization of the pseudorange equations about the errors in the user's estimated *x*, *y*, and *z* coordinates can be performed, as shown in [5]. Equation (8) shows the code phase error expressed as a function of the error in the user's coordinates and clock bias.

$$\varepsilon_i \approx a_{x,i}\delta x_u + a_{y,i}\delta y_u + a_{z,i}\delta z_u + c\delta t_u \quad (8)$$

In equation (8), $a_{x,i}$, $a_{y,i}$, and $a_{z,i}$ are the x, y, and z components of a unit vector pointing from the user's estimated position to the *i*th satellite, respectively. The observation matrix of the EKF is linearized using this technique to accommodate the code phase error measurements, resulting in the observation matrix shown below in Equation (9).

$$H = \begin{bmatrix} a_{x,1} & a_{y,1} & a_{z,1} & 0 & 0 & 0 & 1 & 0 \\ a_{x,2} & a_{y,2} & a_{z,2} & 0 & 0 & 0 & 1 & 0 \\ \vdots & & \vdots & & & \vdots & & \\ a_{x,i} & a_{y,i} & a_{z,i} & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (9)$$

SIMULATION SETUP

The behavior of the VDLL algorithm in response to satellite loss was analyzed using simulated data. The data was generated by a Spirent GPS constellation simulator on Redstone Arsenal. The RF signals produced by the simulator were downconverted, sampled, and saved by using a NordNav RF front-end. The NordNav front-end

uses an Intermediate Frequency (IF) of 4.1304 MHz and a sampling frequency of 16.3676 MHz. A two bit sample resolution was used.

The scenario used for the simulation is a stationary user located in Huntsville, AL. Only the civilian components (C/A PRN code and carrier) of the L_1 GPS signal were tracked. GPS Satellites 16, 5, 3, and 12 were in view and used by the VDLL algorithm. The signals from all the satellites are initially received at a C/No ratio of 40 dB-Hz. At approximately 25 seconds into the simulation, the signals from satellites 16, 5, and 12 are turned off. The signals remain turned off for about 15 seconds. Afterwards, they reappear at their previous C/No ratio of 40 dB-Hz. An integration time of 50 Hz was used throughout the simulation.

Simulated IMU measurements were added to the simulation. The IMU measurements were produced at 50 Hz, synchronous with updates of the position EKF, and the IMU axes were assumed to be aligned with the ECEF coordinate frame. The IMU measurements were modeled as inputs to the velocity states of the EKF. The quality of the IMU measurements is commensurate with that of an IMU400CD [6].

The noise covariance matrix of the EKF was determined empirically and is shown in Equation (10).

$$R = \begin{bmatrix} 75.6142 & 0 & 0 & 0 \\ 0 & 75.6142 & 0 & 0 \\ 0 & 0 & 75.6142 & 0 \\ 0 & 0 & 0 & 75.6142 \end{bmatrix} \quad (10)$$

The measurements of the code phase error produced by the early-minus-late discriminator have a variance of about 75 meters squared. The process noise covariance matrix of the EKF is shown in Equation (11).

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .0139 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .0139 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .0139 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

The diagonal terms corresponding to the position, clock bias, and clock drift states were determined from hand tuning. The terms corresponding to the velocity states were determined from the IMU400CD datasheet, [6].

The noise covariance matrix for the carrier tracking Kalman filters is shown in Equation (12).

$$R = \begin{bmatrix} .0042 & 0 \\ 0 & 35.1508 \end{bmatrix} \quad (12)$$

The terms in equation (12) were determined empirically. The process noise covariance matrix used by the carrier tracking Kalman filter is shown in Equation (13).

$$Q = \begin{bmatrix} .01 & 0 & 0 \\ 0 & .2 & 0 \\ 0 & 0 & .003 \end{bmatrix} \quad (13)$$

The terms of the process noise covariance matrix were determined by hand tuning. The rate of change in the carrier frequency was not predicted, but estimated from the phase and frequency measurements.

For comparison, a traditional Costas loop and DLL were used to process the IF data. Both loops use an integration time of 20 ms, the same as the VDLL. The Costas loop used a noise bandwidth of 10 Hz and a two quadrant arctangent phase discriminator. The DLL used a noise bandwidth of 1 Hz and a normalized early-minus-late discriminator.

SIMULATION RESULTS

Figure 2 shows the output of the in-phase prompt correlator for satellite 16. When the received signals are being tracked, the in-phase prompt output is the amplitude of the received signal multiplied by an integration gain and the sign of the navigation data bit. The VDLL initially tracks the received PRN code. At approximately 27 seconds, the signals from satellite 16 are turned off. At that point, the in-phase prompt output drops below the tracking threshold of the VDLL algorithm. The diagonal terms of the noise covariance matrix for the carrier tracking filter for Satellite 16 are then inflated by the VDLL algorithm. The diagonal term of the noise covariance matrix for the EKF corresponding to the measurement from Satellite 16 is also inflated. Inflating the diagonal terms of the covariance matrices effectively makes the tracking filters ignore measurements.

At approximately 39.3 seconds, the satellite signal returns at its previous C/No level. A close-up of this event is shown in Figure 3. The output of the in-phase prompt signal can then be seen exceeding the tracking threshold. The in-phase prompt value exceeds the threshold because

the replica carrier and PRN code are nearly phase-locked with the received carrier and PRN code, respectively. The carrier tracking filter maintains sufficiently accurate estimates of the carrier signal through the outage to allow the fast reacquisition. Similarly, the phase of the PRN code, predicted by the EKF, is accurate enough to allow the in-phase prompt value to exceed the tracking threshold when the signal reappears.

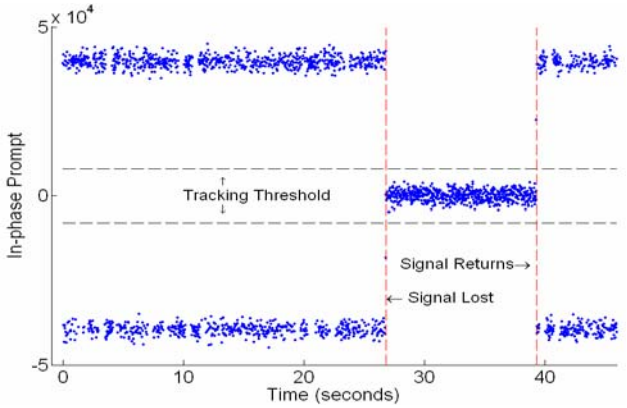

Figure 2: In-Phase Prompt Output by VDLL for SV 16

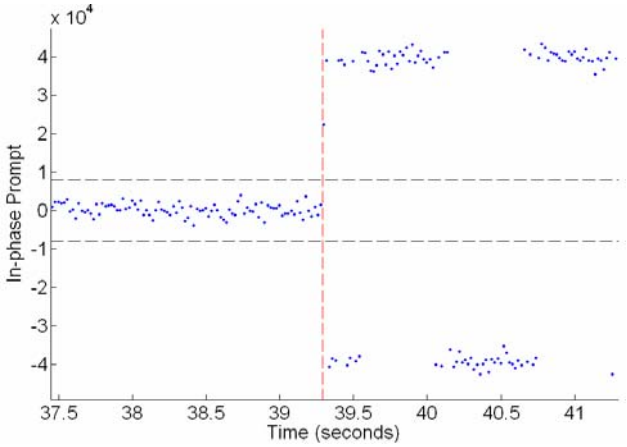

Figure 3: In-Phase Prompt Output by VDLL for SV 16

Figure 4 shows the estimates of the carrier Doppler frequency by the VDLL algorithm and the traditional Costas loop. During the satellite outage, the carrier tracking filter ignores the phase and frequency measurements. The filter propagates forward its estimates of the carrier phase, frequency, and rate of frequency change during the outage. Its estimates are sufficiently accurate to allow the in-phase prompt value to exceed the tracking threshold when the signal returns. The noise covariance matrix is reset when the signal returns and the carrier tracking filter begins considering the measurements again. The traditional Costas loop's estimate of the carrier frequency diverges during the outage. When the signal returns, the Costas loop does not lock to the carrier signal.

Figure 5 shows the output of the code phase discriminator for the VDLL and the traditional DLL. During the signal outage, the discriminator operates on noise and the variance of the measurements is significantly larger. When the signal returns, the code phase predicted by the EKF is within the linear operating range of the code discriminator. The code phase estimate maintained by the DLL is not accurate enough to allow it to relock to the PRN code.

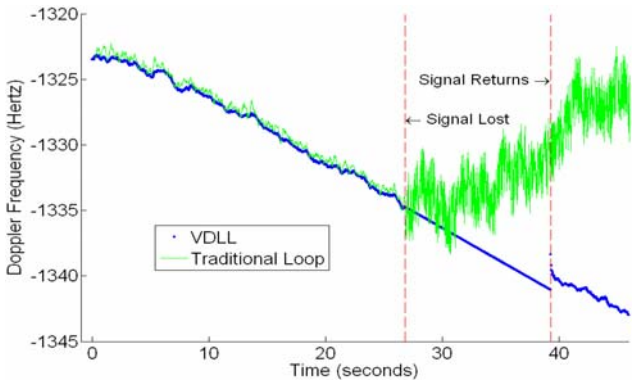

Figure 4: Doppler Frequency Estimates by VDLL and Costas loop for SV 16

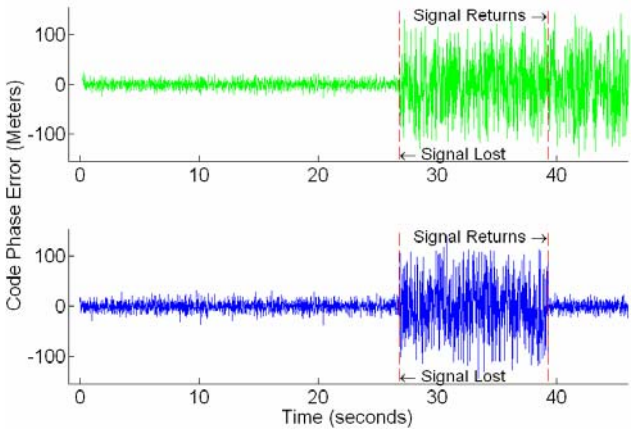

Figure 5: Code Discriminator Output for SV 16

The VDLL is able to quickly reacquire the signal for Satellite 16 when it reappears. The VDLL performs in a very similar manner for Satellites 5 and 12 when the signals coming from them are lost and then returned. The only satellite signal that is uninterrupted during the simulation is that from Satellite 3. Figure 6 shows the in-phase prompt correlator output for Satellite 3. As can be seen, the VDLL tracks the carrier and PRN signals throughout the simulation.

Figures 7, 8, and 9 show the errors in the EKF's estimates of the user's x, y, and z coordinates, respectively. The EKF is initialized with errors in the three coordinates. The EKF's estimates can be seen to rapidly converge towards their correct values. At approximately 26 seconds into the simulation, three of the four satellite signals are lost. The

diagonal terms of the EKF relating to the measurements from those satellites are inflated at that time. The EKF effectively only has one measurement during the outage. The x, y, and z estimates remain nearly static throughout the period of the blockage. The EKF places the residual from the measurements in the clock drift state, shown in Figure 10. This is due to the presence of the IMU inputs. The process noise associated with the clock states is larger relative to the process noise in the velocity and position states.

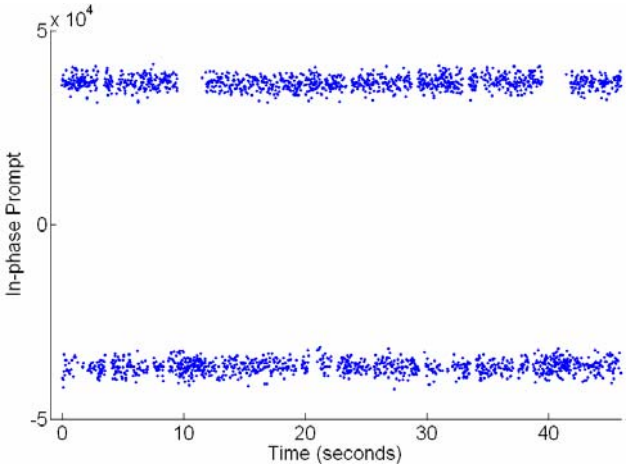

Figure 6: In-Phase Prompt Output by VDLL for SV 12

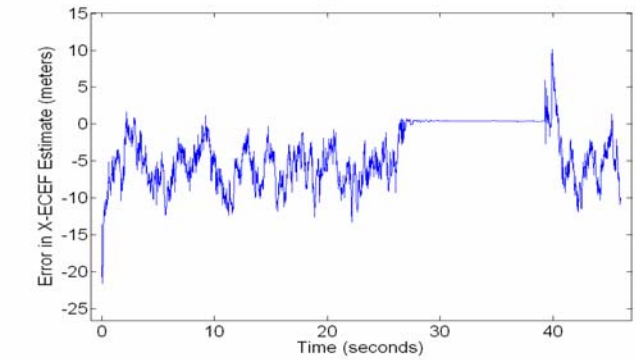

Figure 7: EKF Estimate of X-ECEF Coordinate

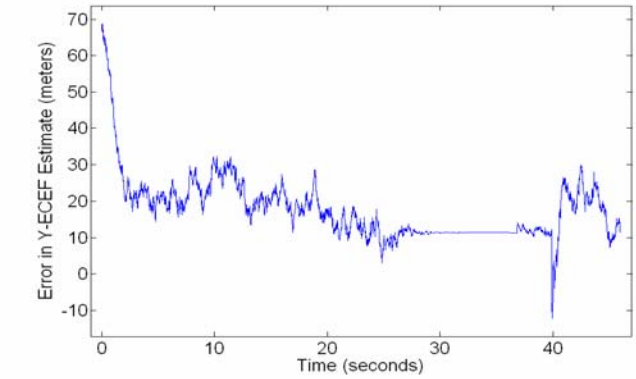

Figure 8: EKF Estimate of Y-ECEF Coordinate

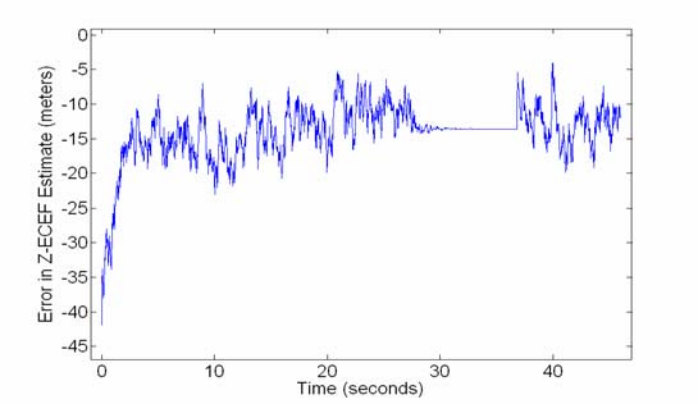

Figure 9: EKF Estimate of Z-ECEF Coordinate

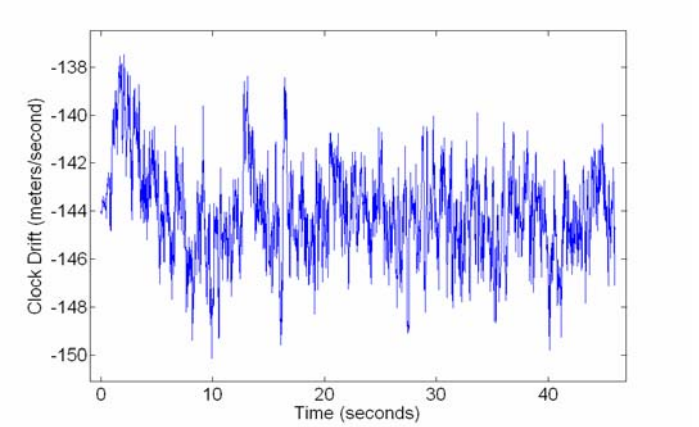

Figure 10: EKF Estimate of Clock Drift

CONCLUSION

In this paper, a realization of the vector delay lock loop concept was introduced. The algorithm introduced in the paper is based on nonlinear discriminators. It uses a single extended Kalman filter to track the satellite PRN sequences and the user's position, velocity, and clock states. A Kalman filter is used in each channel to track the individual satellite carrier signals. The performance of the VDLL algorithm was analyzed using simulated data. Specifically, the ability of the VDLL to reacquire blocked signals was investigated. For purposes of comparison, traditional tracking loop based methods were used on the same data. The VDLL was seen to rapidly reacquire the blocked satellite signals when they reappeared. The traditional methods did not reacquire the blocked signals once they reappeared.

There is great potential for future work in the area of vector tracking. For the particular algorithm described in this paper, future work includes extending the concept of vector tracking to the satellite carrier signals. The performance of the algorithm in various environments, such as high noise and high dynamic environments, has yet to be fully explored. Coupling the vector tracking

algorithm with a full inertial navigation system would also be included in future work.

ACKNOWLEDGMENTS

The authors wish to thank the personnel of the Aviation and Missile Research, Development, and Engineering Center in Huntsville, AL for funding and guidance of this work.

REFERENCES

[1] Spilker, J.J. *Fundamentals of Signal Tracking Theory*. In : Global Positioning System: Theory and Applications, Vol. I. Progress in Astronautics and Aeronautics, Volume 163, AIAA, Washington, DC, 1996. St. Petersburg, Russia

[2] R. F. Stengel, *Optimal Control and Estimation*. New York: Dover Publications, 1994.

[3] M. Sayre, "Development of a Block Processing Carrier to Noise Ratio Estimator for the Global Positioning System," M.S. thesis, Ohio University, Athens, OH, United States, 2003.

[4] Bullock J. B., M. Foss, G. J. Geier, and M. King, *Integration of GPS with Other Sensors and Network Assistance*. In : Understanding GPS: Principles and Applications, Second Edition, Mobile Communication Series, Chapter 9, pages 459-558. Artech House Publishers

[5] Kaplan E. D., J. L. Leva, D. Milbert, and M. S. Pavloff, *Fundamentals of Satellite Navigation*. In : Understanding GPS: Principles and Applications, Second Edition, Mobile Communication Series, Chapter 2, pages 21-65. Artech House Publishers

[6] Crossbow Technology (2006). IMU400CD Data Sheet.