Comparison of Traditional Tracking Loops and Vector Based Tracking Loops for Weak GPS Signals

Matthew Lashley, *Navigation Technology Associates*
David M. Bevly, *Auburn University*

BIOGRAPHY

Matthew Lashley is an employee of Navigation Technology Associates, Inc. and a member of the GPS and Vehicle Dynamics Lab (GAVLAB) at Auburn University. 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 associate professor at Auburn University and director 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/Frequency Lock Loop (VDFLL) algorithm is introduced for the GPS L_1 civilian signal. The VDFLL algorithm uses a single Extended Kalman Filter (EKF) to simultaneously track the GPS signals and determine the user's navigation states. The states of the EKF are the user's position, velocity, acceleration, clock bias, and clock drift. The carrier frequency and Pseudo-Random Noise (PRN) code phase for each satellite are predicted based on the states of the EKF. Unlike traditional approaches, the VDFLL algorithm does not use Delay Lock Loops (DLL's) or Costas Loops to track the GPS signals. The VDFLL algorithm has the ability to track weak GPS signals and rapidly reacquire blocked signals. The performance of the VDFLL is compared to that of a high end commercial receiver in an environment with dense foliage and rapidly fluctuating signal levels. The experimental results show that the VDFLL outperforms the commercial receiver and provides continuous coverage in GPS challenged environment.

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 DLL is used to track the 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. Additionally, 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/Frequency Lock Loop combines the tracking of the different satellite PRN and carrier signals into a single algorithm, as originally introduced in [1]. The VDFLL operates on two simple principles. First, the phases of the received PRN sequences are based on the user's position. Second, the Doppler frequencies of the

received carrier signals are based on the user's velocity. So, given the satellite ephemeris, the user's position, velocity, clock bias, and clock drift, the PRN sequences and carrier frequencies from each satellite can be predicted. The VDFLL uses estimates of the user's position and clock bias to generate the replica PRN sequences. The replica carrier signals are generated by using the estimates of the user's velocity and clock drift. The error signals generated after each integration interval are then used to estimate the user's position, velocity, clock bias, and clock drift. An Extended Kalman Filter [2] is used to track the user's navigation states (position, velocity, clock bias, and clock drift) and process the error signals generated after every integration interval, as described in [3], [4]. In a VDFLL, the tracking of all the received signals is handled by the single EKF.

The VDFLL has several potential advantages over the traditional tracking loop approach. First, the EKF can weight measurements of the code phase and carrier frequency errors. 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 signals being controlled by the EKF's estimate of the user's navigation states. Third, the VDFLL has the potential to rapidly reacquire signals after a satellite blockage.

VDFLL ARCHITECTURE

The VDFLL algorithm presented in this paper uses a single EKF to track the received signals and determine the user's navigation states. The user's navigation states are position, velocity, acceleration, clock bias, and clock drift. The code phases and carrier frequencies are predicted based on the EKF's states. The code and carrier Numerically Controlled Oscillators (NCO's) in each channel are initialized with the predictions at the beginning of each integrate and dump operation. At the end of each integrate and dump operation, the correlator outputs from each channel are used as measurements for the EKF. Using the correlator outputs as measurements constitutes closing a feed back loop through the EKF. Figure 1 shows a graphical representation of the VDFLL architecture.

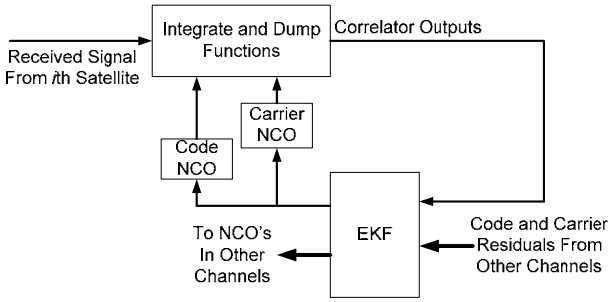

Figure 1: Vector Delay/Frequency Lock Loop Architecture

The states and discrete time state transition matrix of the EKF are shown in Equation (1). The EKF actually tracks the errors in the user's estimated states [5]. The estimated states themselves are kept track off outside of the filter. The first six states represent the errors in the user's estimated position, velocity, and acceleration in the Earth-Centered Earth-Fixed (ECEF) coordinate frame, respectively. The last two states are errors in the user's estimated clock bias and clock drift. The terms *a* and *b* are functions of the time interval (ΔT) between measurement updates of the filter.

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

$$a = \Delta T \quad b = \frac{\Delta T^2}{2} \quad \Delta T = \text{Time between filter updates}$$

The acceleration states and clock drift states are modeled as static in Equation (1). The diagonal terms of the process noise covariance matrix corresponding to the acceleration and clock drift states were hand tuned to allow the filter to track changes in the user's acceleration and clock drift.

The code phase predictions for each channel are based on the user's estimated position and clock bias and the position of the satellite being tracked in the specific channel. Predicting the code phases corresponds to predicting the pseudoranges for each satellite. The predicted pseudorange for the *i*th channel is shown in Equation (2) [6].

$$\hat{\rho}_i = \sqrt{(\hat{x}_u - x_i)^2 + (\hat{y}_u - y_i)^2 + (\hat{z}_u - z_i)^2} + c\hat{t}_u \quad (2)$$

In Equation (2), 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.

The frequency of the received carrier in each channel is predicted using Equation (3), [6]. In (3), $\hat{\dot{x}}_u$, $\hat{\dot{y}}_u$, $\hat{\dot{z}}_u$, and $\hat{\dot{t}}_u$ are the user's estimated velocity and clock drift, respectively. The term v_i is the velocity vector of the *i*th satellite and a_i is the unit vector from the user to the *i*th satellite, both of these vectors are in the ECEF frame.

$$\hat{f}_i = \frac{\left\{1 - \frac{1}{c}\left[v_i - \left\langle \hat{\dot{x}}_u, \hat{\dot{y}}_u, \hat{\dot{z}}_u \right\rangle \bullet a_i\right]\right\}}{\left(1 + \hat{\dot{t}}_u\right)} \quad (3)$$

The code and carrier NCO's for each channel are initialized with the predictions generated from Equations (2) and (3) at the beginning of each integrate and dump period. An integration time of 20 ms is used for all channels. 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 [7], [8]. In addition to the six correlator outputs shown above, several "noise correlator" outputs are generated by initializing the code NCO with an offset in one of the nulls of the Gold code autocorrelation function. These "noise correlator" samples are used to generate an estimate of the noise power for determining the C/No ratio. This technique is described in [9].

The correlator outputs are used to generate estimates of the error between the predicted and received PRN code phases. A normalized early-minus-late discriminator [11] is used to produce measurements of the code phase error from the correlator outputs, as shown in Equation (4).

$$\varepsilon \approx \frac{\Sigma\sqrt{IE^2 + QE^2} - \Sigma\sqrt{IL^2 + QL^2}}{\Sigma\sqrt{IE^2 + QE^2} + \Sigma\sqrt{IL^2 + QL^2}} \quad (4)$$

The term ε is a measurement of the difference between the true distance between the user and the satellite, and the predicted distance using the EKF's estimates of the user's position.

Likewise, a measurement of the error between the predicted and received carrier frequencies is generated using the formulas shown below, [11].

$$\begin{aligned} dot &= IP_{t1} \cdot IP_{t2} + QP_{t1} \cdot QP_{t2} \\ cross &= IP_{t1} \cdot QP_{t2} + IP_{t2} \cdot QP_{t1} \\ f_{error} &\approx \frac{\arctan\left(\frac{cross}{dot}\right)}{360(t_2 - t_1)} \end{aligned} \quad (5)$$

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. The frequency error measurement generated in Equation (5) is scaled to convert it to a velocity error measurement, shown in Equation (6).

$$v_{error} = f_{error} \cdot \frac{c}{f_{L_1}} \quad (6)$$

In Equation (6), *c* is the speed of light and f_{L_1} is the nominal L_1 carrier frequency. The term v_{error} is a measurement of the difference between the true relative velocity between the user and the satellite and the relative velocity calculated from the EKF's estimates.

In addition to the code phase and carrier frequency error measurements, the amplitude of the received is measured using the formula shown in Equation (7), [11].

$$\tilde{A} = \sqrt{IP^2 + QP^2} \quad (7)$$

The amplitude estimates and noise power estimates are processed to smooth them. The C/No ratio is calculated for each channel using the smoothed noise power estimate and the smoothed amplitude estimates, as described in [9].

The code phase error (4) and velocity error (6) are used as measurements for the EKF. The measurements from each

channel arrive at different times and are processed sequentially. The observation matrix of the EKF is linearized at each epoch to accommodate the error measurements from each channel, this results in the observation matrix shown below in Equation (9).

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

The derivation of this linearization can be found in [6].

The measurement noise covariance matrix of the EKF for each channel is determined from the C/No estimates. The variance of the code phase and frequency error measurements versus C/No level was determined empirically and saved as a look up table.

TEST SETUP

The ability of the VDFLL algorithm to track weak GPS signals was demonstrated using data collected from Auburn University's Mini Baja racing course. The Mini Baja course is a narrow race track through an area of dense foliage. The majority of receivers that have been used to navigate about the course have experienced significant periods of GPS outages. A NovAtel ProPak-V3 receiver was used as a comparison for the VDFLL. Both receivers were mounted on an All Terrain Vehicle (ATV) and driven around the Mini Baja course. The NovAtel and VDFLL data were collected on different runs around the Mini Baja course, but within a few minutes of each other. The path taken by the ATV when collecting data for the VDFLL was virtually identical to the path driven when collecting NovAtel data.

For the VDFLL, the RF signals from the satellites 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 in the IF collection. The sampled data was post-processed using MATLAB.

TEST RESULTS

At the time the data was recorded, there were six GPS satellites in view. Satellite's 6, 27, 11, 28, 17, and 19 were used by the VDFLL. The ATV was stationary for approximately the first 100 seconds of the data set. It took the ATV around 842 seconds to complete the Mini Baja course. The vehicle returned to its starting point and remained static for the last 252 seconds of the data set.

The tracking results from the VDFLL for each satellite are shown in Figures 2 through 7. Each figure shows the pseudorange error, frequency error, and estimated C/No level for each satellite. The pseudorange and frequency errors shown are the measurements produced at the end of each integrate and dump operation. At the start of the run, the satellite C/No ratios were at nominal levels (~45 dB-Hz). The C/No ratios can be seen to rapidly fluctuate when the ATV is navigating the Mini Baja course. A threshold of 23 dB-Hz was set for declaring a satellite blocked. The only satellite that was declared blocked was satellite 19. This is most likely due to the fact that satellite 19 was at an elevation of 16.5º. The heavy foliage and hilly terrain of the course therefore affected its signal more than the other satellites.

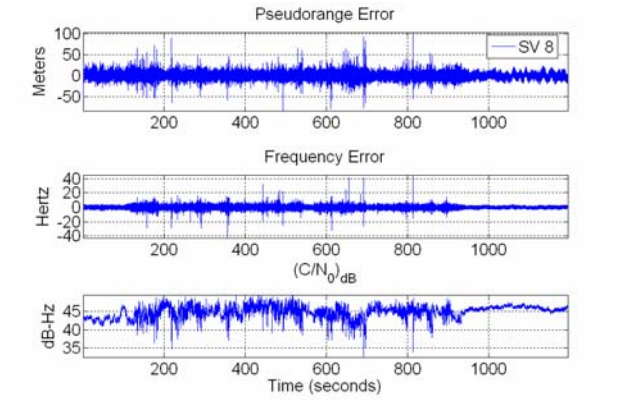

Figure 2: VDFLL Tracking Results for Satellite 8

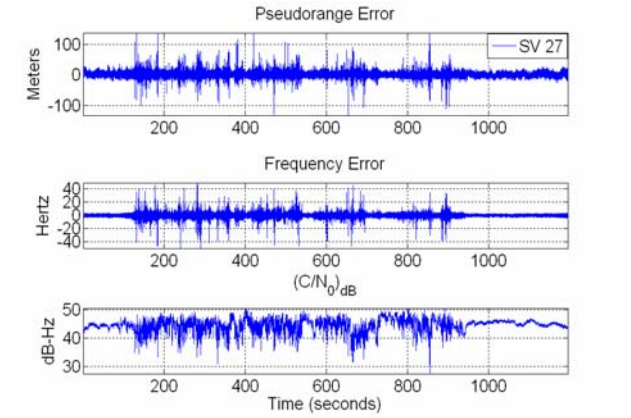

Figure 3: VDFLL Tracking Results for Satellite 27

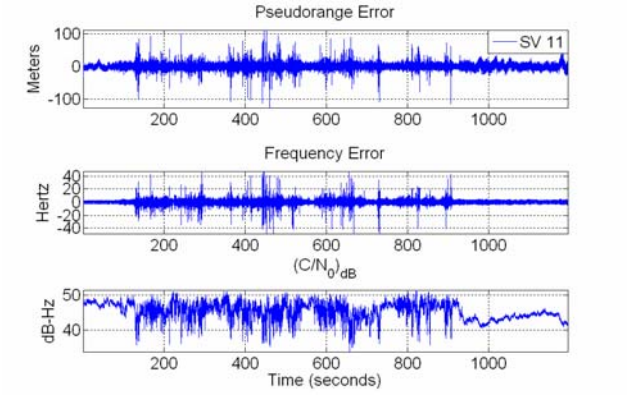

Figure 4: VDFLL Tracking Results for Satellite 11

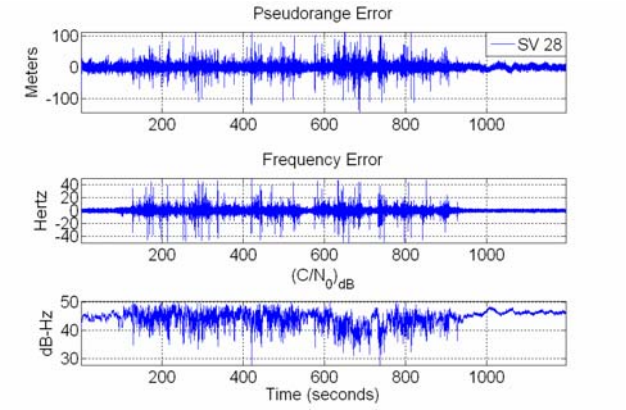

Figure 5: VDFLL Tracking Results for Satellite 28

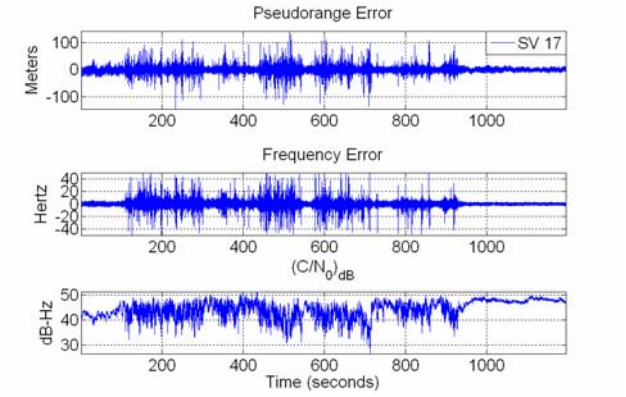

Figure 6: VDFLL Tracking Results for Satellite 17

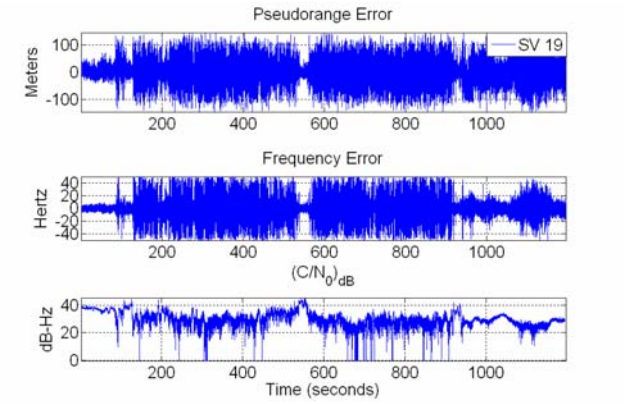

Figure 7: VDFLL Tracking Results for Satellite 19

The position solutions from the VDFLL algorithm and the NovAtel receiver are plotted together in Figure 8. The positions have been converted from the ECEF frame into a local East-North-Up (ENU) coordinate frame. The East and North components are plotted to demonstrate the difference in coverage between the VDFLL and NovAtel receivers. Figures 9 through 12 present portions of the data in Figure 8, but split into smaller graphs to better show the coverage from each receiver. The figures demonstrate that there are multiple stretches of the track where the NovAtel receiver has no GPS position solution. For example, in Figure 10 there is a large gap between East (-730, -685) and North (75, 92). On the other hand, the VDFLL provides continuous coverage throughout the entire course. Additionally, there are multiple aberrant position solutions from the NovAtel receiver. Two of these solutions can be seen clearly in Figure 10 at the lower right side of the graph.

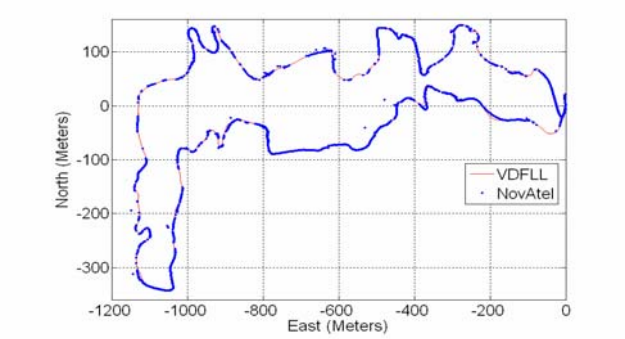

Figure 8: VDFLL and NovAtel East-North Positioning Results

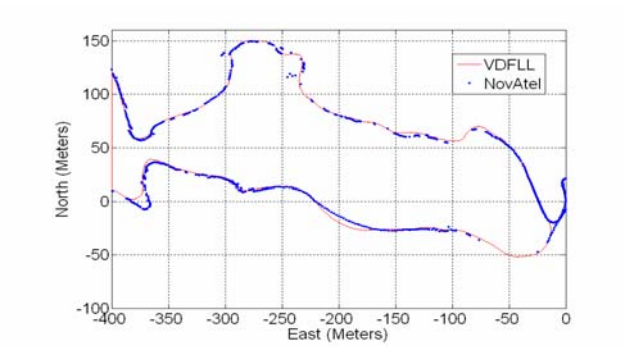

Figure 9: VDFLL and NovAtel East-North Positioning Results

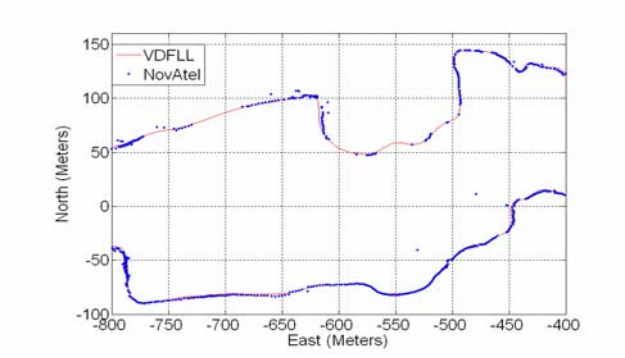

Figure 10: VDFLL and NovAtel East-North Positioning Results

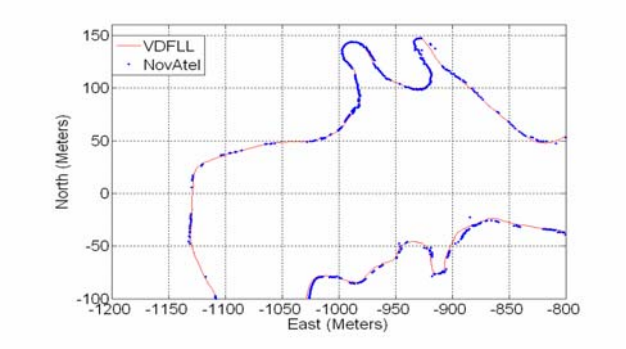

Figure 11: VDFLL and NovAtel East-North Positioning Results

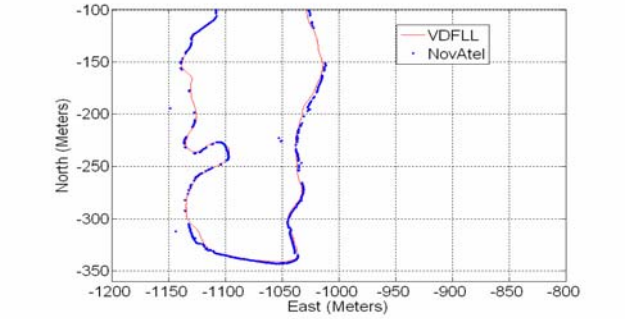

Figure 12: VDFLL and NovAtel East-North Positioning Results

CONCLUSION

In this paper, a variant of the Vector Delay/Frequency Lock Loop algorithm was developed for the GPS L_1 civilian signal. It uses a single extended Kalman filter to track the satellite signals and the user's position, velocity, acceleration, and clock states. The EKF's states are used to predict the received signals from each satellite. The algorithm then uses nonlinear discriminator functions to turn the correlator outputs, produced at the end of each integration period, into linear measurements of the code phase and carrier frequency error. These measurements are used to estimate errors in the user's navigation states.

The performance of the VDFLL algorithm in a GPS challenged environment was compared to that of a high end commercial receiver. The VDFLL provided continuous coverage through dense foliage. The commercial receiver had multiple periods during which there was no GPS coverage.

There is great potential for future work in the area of vector tracking. The performance of the algorithm in various environments, such as high noise and high dynamic environments has yet to be fully explored. Adding an Inertial Measurement Unit (IMU) to the system would improve its performance in high dynamic environments and ability to reacquire signals after longer outages. Merging an IMU with the VDFLL would constitute a Deeply Integrated (DI) GPS-INS algorithm. Also, the ability of the VDFLL to function in the presence of intentional jamming has not been explored.

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. The authors would also like to thank Kenneth Lambert and William Travis for aiding in collecting the presented data.

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] T. Pany, R. Kaniuth, and B. Eissfeller, "Deep Integration of Navigation Solution and Signal Processing," in *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2005* , 2005, pp. 1095-1102

[4] M. G. Petovello, and G. Lachapelle, "Comparison of Vector-Based Software Receiver Implementations with Applications to Ultra-Tight GPS/INS Integration," in *Proceedings of the 19th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2006* , 2006, pp. 1790-1800

[5] 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

[6] 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

[7] M. L. Psiaki and H. Hung, "Extended Kalman Filter Methods for Tracking Weak GPS Signals," in *Proceedings of ION GPS 2002,* 2002, pp. 2539-2553

[8] 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.

[9] R. N. Crane, "A Simplified Method for Deep Coupling of GPS and Inertial Data," in *Proceedings of the 2007 National Technical Meeting of the Institute of Navigation*, 2007, pp. 311-319

[10] Ward P. W., J. W. Betz, and C. J. Hegarty, *Satellite Signal Acquisition, Tracking, and Data Demodulation*. In : Understanding GPS: Principles and Applications, Second Edition, Mobile Communication Series, Chapter 5, pages 153-241. Artech House Publishers

[11] N. I. Ziedan, *GNSS Receivers for Weak Signals*. Boston: Artech House, 2006.