

Article

# Design and Implementation of Vector Tracking Loop for High-Dynamic GNSS Receiver

Rongjun Mu and Teng Long \* 

School of Astronautics, Harbin Institute of Technology, Harbin 150001, China; murjun@163.com

\* Correspondence: 16B918074@stu.hit.edu.cn

**Abstract:** For the tracking of high-dynamic satellite navigation signals, the conventional scalar tracking loop (STL) is vulnerable. Frequent signal-tracking interruption affects the continuity of navigation. The vector tracking loop (VTL) can overcome this disadvantage. However, there are some difficulties in implementing existing vector tracking methods on a real-time hardware receiver, such as the synchronization problem and computation load. This paper proposes an implementation framework of VTL based on a partial open-loop numerically controlled oscillator (NCO) control mode that can be implemented with minor modifications on an existing receiver platform. The structure of VTL, the design of the navigation filter, and the key points of hardware implementation are introduced in detail. Lastly, the VTL performance was verified by a GPS simulator test. The results show that the proposed VTL can run in real-time and be significantly improved in the tracking continuity of high-dynamic signals, tracking sensitivity, positioning accuracy, and recovery time for interrupted signals compared with those of STL.

**Keywords:** GNSS; vector tracking; high-dynamic; hardware receiver



**Citation:** Mu, R.; Long, T. Design and Implementation of Vector Tracking Loop for High-Dynamic GNSS Receiver. *Sensors* **2021**, *21*, 5629. <https://doi.org/10.3390/s21165629>

Academic Editors: Kamil Krasuski and Chris Rizos

Received: 19 June 2021

Accepted: 18 August 2021

Published: 20 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Global navigation satellite systems (GNSS) are widely used in positioning and navigation tasks for artificial flight vehicles as a high-availability, high-precision, and low-cost positioning technology. Some of these vehicles have high velocity and acceleration, such as missiles, rockets, and hypersonic aircraft. This requires the GNSS receiver to have the ability to track high-dynamic signals. As early as 1988, the Jet Propulsion Laboratory (JPL) researched high-dynamic global positioning system (GPS) signal tracking according to the application requirement of the GPS Range Application Joint Program Office (RAJPO) [1]. In that report, a scenario with an acceleration of 50 g and a jerk of 100 g/s was proposed as the upper bound for the study of high-dynamic signal tracking.

There are three main factors affecting the high-dynamic performance of a receiver: Loop filter characteristics, oscillator noise, and oscillator vibration sensitivity [2]. The latter two can be improved on the hardware-design level [3], but the first needs to be improved on the software and algorithmic levels, which are the focus of this paper. There are many proposed methods on how to achieve high-dynamic signal tracking, such as second-order frequency lock loop (FLL) assisted third-order phase lock loop (PLL) [4], Kalman filter-based [5], maximum likelihood-based [6], and fractional Fourier transform-based [7,8]. However, these methods focus only on a single channel and ignore overall performance. Subject to the nature of scalar tracking loop (STL) in which signal-tracking channels are independent of each other, the optimization of one channel cannot optimize the receiver. Some channels may still lose the lock when a high-dynamic or weak signal is encountered. The problem of the continuity of signal tracking has not been solved. Fortunately, the vector tracking loop (VTL) was proposed to overcome the disadvantage of STL [9]. The principle of VTL is to combine the signal tracking of all channels via the receiver's position, velocity, and time (PVT) solution, which gives several advantages. First, information fusion

between channels can reduce noise and keep the error of the local replica signal small enough to stay within the linear region of tracking [10]. Second, strong signals can assist in weak-signal tracking, and low-dynamic signals can assist in high-dynamic-signal tracking. Third, when parts of signals are interrupted, a VTL can bridge these signals through the PVT solution [11]. In theory, vector tracking is an ideal framework for designing a satellite navigation receiver.

There are two types of VTL, centralized and cascaded. The centralized VTL is rarely used on account of its computational complexity and nonlinearity [12]. The cascaded VTL, which is widely studied, can be summarized into a unified framework [13], as shown in Figure 1. The basic principle of this framework is that the code phase and carrier frequency of each channel, which is used to control code and carrier numerically controlled oscillator (NCO), are calculated by the receiver's PVT solution. The PVT solution is corrected by the navigation errors that are estimated by the navigation filter according to the error of the code phase and the carrier frequency, which are estimated by the signal tracking error estimator. Within the unified framework, the difference between various vector tracking methods is implementing the signal tracking error estimators, such as discriminator-based [14] and prefilter-based [10,15] estimators. Most researchers implement a VTL using a software-defined GNSS receiver (SDGR) [16–18]. In SDGR, a digital intermediate-frequency (IF) signal is collected and then batch processed in a general-purpose processor (GPP) or a digital signal processor (DSP). Signal processing is flexible, so that the code phase and carrier phase of the local replica signal can be freely controlled. The unified framework is easy to implement in an SDGR. However, in a GNSS hardware receiver, NCO and correlator run at high speed in a special integrated circuit or a field-programmable gate array (FPGA), so it is hard to accurately control the code phase at a certain moment. Therefore, the framework needs to be modified to fit hardware implementation. An implementation framework of VTL based on a partial open-loop NCO control mode is proposed in this paper. In this framework, the VTL can easily be implemented using the existing correlator and NCO control interface, and it has the ability of high-dynamic signal tracking.

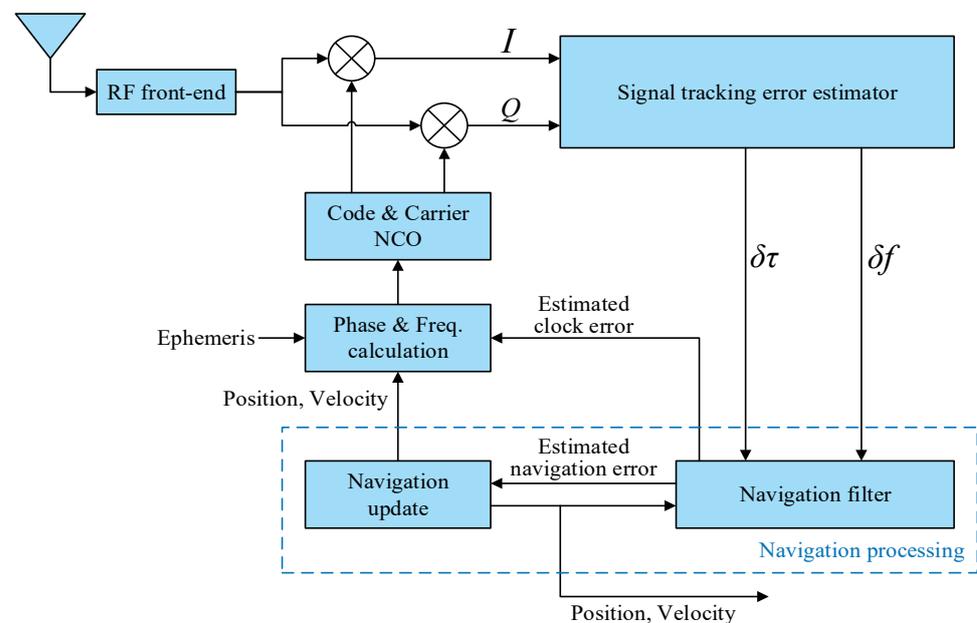


Figure 1. The unified framework of vector tracking loop.

The remainder of the paper is organized as follows. Section 2 introduces our methods, including the structure and principle of the proposed VTL implementation framework, the method of navigation processing, a processing strategy of weak-signal channels, and the structure and key design points of the vector tracking hardware receiver. In Section 3, the

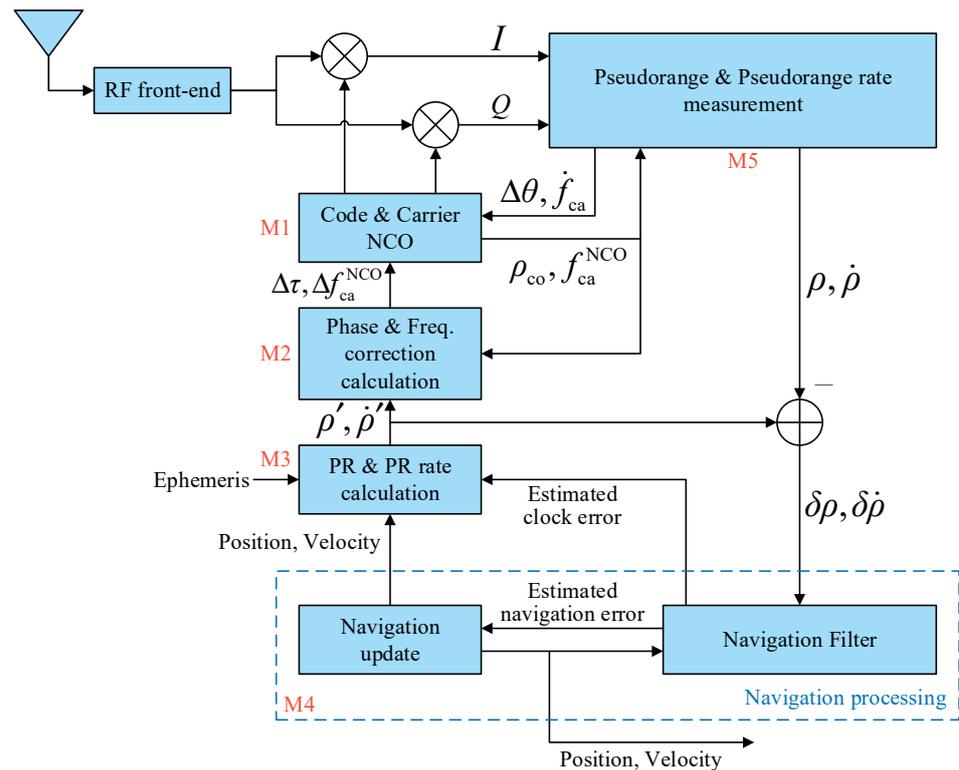
results of high-dynamic experiments based on a GPS simulator are analyzed. Section 4 discusses some future research directions of VTL. Section 5 summarizes the study.

## 2. Methods

### 2.1. Vector Tracking Loop

#### 2.1.1. Loop Structure

The diagram of the proposed implementation framework of VTL based on a partial open-loop NCO control mode is shown in Figure 2.



**Figure 2.** Implementation framework of vector tracking loop.

Symbols in Figure 2:

M1–M5 the module number, which will be referenced below;

$\rho_{co}$  code pseudorange, which is calculated by the code phase;

$f_{ca}^{NCO}$  carrier NCO frequency;

$\dot{f}_{ca}$  estimated carrier frequency change rate;

$\Delta\tau$  code phase correction value;

$\Delta f_{ca}^{NCO}$  carrier NCO frequency correction value;

$\Delta\theta$  carrier phase correction value;

$\rho, \dot{\rho}$  measured pseudorange and pseudorange rate;

$\rho', \dot{\rho}'$  calculated pseudorange and pseudorange rate;

$\delta\rho, \delta\dot{\rho}$  pseudorange error and pseudorange rate error.

Code and carrier NCO (M1) generates local replica code and carrier that are phase-aligned with the real signal. In the unified framework of VTL, NCOs run in full closed-loop mode; this means that each NCO control command is computed by the PVT solution. The term “partial open-loop” refers to NCOs running in open-loop mode during the navigation update interval, and then being corrected after the navigation update. During open-loop tracking, the NCO control command does not come from the PVT solution, but from the channel’s own information. The open-loop NCO control command requires high accuracy, that is, the open-loop error being less than the closed-loop error. To achieve this, carrier

NCO frequency  $f_{ca}^{NCO}$  is used to calculate code NCO frequency  $f_{co}^{NCO}$ . Because of the strict linear relationship between the carrier frequency and code frequency, and code frequency being much less than radio-frequency (RF) carrier frequency, the calculated code frequency is so accurate that the code phase drift is very small in the short term. The calculation formula of  $f_{co}^{NCO}$  is

$$f_{co}^{NCO} = \left( 1 + \frac{f_{ca}^{NCO} - f_{IF}}{F_{RF}} \right) F_{co} \quad (1)$$

where  $f_{IF}$  is the IF frequency after down-conversion,  $F_{RF}$  is the RF carrier center frequency,  $F_{co}$  is the nominal code frequency (e.g., GPS L1 C/A signal  $F_{RF} = 1575.42$  MHz,  $F_{co} = 1.023$  MHz). For carrier open-loop tracking, carrier frequency change rate  $\dot{f}_{ca}$  is used to control carrier NCO frequency update so as to track the change in carrier doppler caused by the vehicle motion.  $\dot{f}_{ca}$  comes from the pseudorange rate measurement module (M5); this is explained in detail in Section 2.1.2. The update process of  $f_{ca}^{NCO}$  is

$$f_{ca}^{NCO+} = f_{ca}^{NCO-} + \dot{f}_{ca} T_{NCO} \quad (2)$$

where the superscript  $-$  denotes before update, superscript  $+$  denotes after update,  $T_{NCO}$  is the carrier NCO update period.

Correction information comes from the optimal estimation of the PVT solution by the navigation processing module (M4). Since the NCO is not directly controlled by the PVT solution during open-loop tracking, the measurement extracted from the channel is not tracking error but pseudorange and pseudorange rate (M5). The code pseudorange and carrier NCO frequency represent the current state of open-loop tracking, the pseudorange and pseudorange rate calculated by the estimated PVT solution (M3) are the expected state of signal tracking, the difference between the current and expected state results in the code phase correction value  $\Delta\tau$  and carrier NCO frequency correction value  $\Delta f_{ca}^{NCO}$  (M2):

$$\Delta\tau = -\frac{\rho' - \rho_{co}}{\lambda_{co}} \quad (3)$$

$$\Delta f_{ca}^{NCO} = -\frac{\dot{\rho}'}{\lambda_{ca}} - \left( \dot{f}_{ca}^{NCO} - \dot{f}_{IF} \right) \quad (4)$$

where  $\lambda_{co}$  is the code length,  $\lambda_{ca}$  is the RF wavelength. Due to the influence of the receiver clock drift, the relationship between the frequency of the digital signal after A/D sampling and the real signal is

$$f_{real} = (1 + df_r) f_{digital} \quad (5)$$

where  $df_r$  is the receiver clock drift, its unit is usually ppm. Therefore, more accurately, Equation (4) should be written as

$$\Delta f_{ca}^{NCO} = -\frac{\dot{\rho}'}{\lambda_{ca}(1 + df_r)} - \left( \dot{f}_{ca}^{NCO} - \dot{f}_{IF} \right) \quad (6)$$

In partial open-loop NCO control mode, the tracking of each channel is independent during open-loop tracking, and the correction process couples all channels together to achieve vector tracking. Because the correction value is the increment in the current internal state of the NCO, it does not care about the absolute value of the code phase and carrier NCO frequency, there is no special requirement for the time to perform these corrections.

### 2.1.2. Pseudorange and Pseudorange Rate Measurement

The principle of pseudorange and pseudorange rate measurement is shown in Figure 3a.



By organizing Equation (10) and simplifying some symbols, update equations of the carrier frequency estimator can be obtained:

$$\Delta f_{ca} = f_{ca}^{NCO} - f_{ca}^- \quad (11)$$

$$\delta\theta_0 = \delta\theta + \Delta f_{ca} T_{coh} \quad (12)$$

$$\Delta\theta = \alpha \cdot \delta\theta_0 - \Delta f_{ca} T_{coh} \quad (13)$$

$$f_{ca}^+ = f_{ca}^- + \beta \cdot \delta\theta_0 \quad (14)$$

$$\dot{f}_{ca}^+ = \dot{f}_{ca}^- + \gamma \cdot \delta\theta_0 \quad (15)$$

Essentially, the carrier frequency estimator is a digital third-order PLL. The structure of the third-order loop filter is shown in Figure 3b.  $\Delta f_{ca} T_{coh}$  is the additional carrier phase error caused by the difference between the NCO frequency and the estimated frequency.  $\delta\theta_0$  is the carrier phase error input to the PLL. Because the effect of  $\Delta f_{ca} T_{coh}$  has been removed from the directly measured carrier phase error, the estimation result of  $f_{ca}$  does not change even if  $f_{ca}^{NCO}$  is corrected externally. This is the basis of the vector carrier tracking that the carrier frequency estimation is independent of the carrier NCO control.  $\Delta\theta$  is used to accomplish carrier phase locking.  $\dot{f}_{ca}$  is provided to the carrier NCO for open-loop update. The value of  $\alpha, \beta, \gamma$  can be determined by typical parameters of third-order PLL [19], as follows:

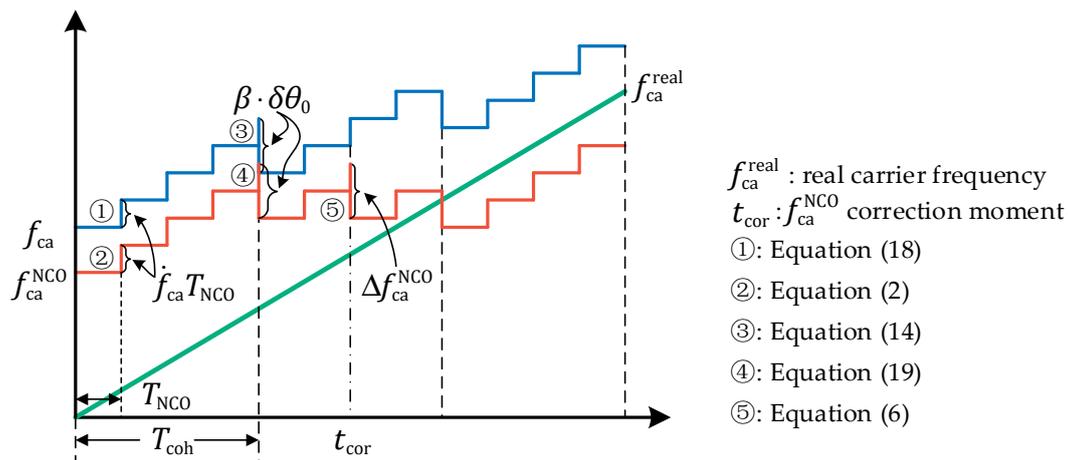
$$\begin{cases} \alpha = K_1 T_{coh} = 2.4\omega_0 T_{coh} \\ \beta = K_2 T_{coh} = 1.1\omega_0^2 T_{coh} \\ \gamma = K_3 T_{coh} = \omega_0^3 T_{coh} \end{cases} \quad (16)$$

$$\omega_0 = 1.275B_n \quad (17)$$

where  $B_n$  is the equivalent noise bandwidth, and  $\omega_0$  is the natural frequency.

Figure 4 shows the update process of  $f_{ca}$  and  $f_{ca}^{NCO}$  more visually. Note that  $f_{ca}$  needs to be updated synchronously with  $f_{ca}^{NCO}$  after each carrier NCO update:

$$f_{ca}^+ = f_{ca}^- + \dot{f}_{ca} T_{NCO} \quad (18)$$



**Figure 4.** Diagram of the update process of  $f_{ca}$  and  $f_{ca}^{NCO}$ . (For more clearly, a coherent integration time contains four carrier NCO update periods.).

And  $f_{ca}^{NCO}$  needs to increase the same increment with  $f_{ca}$  when the carrier frequency estimator update:

$$f_{ca}^{NCO+} = f_{ca}^{NCO-} + \beta \cdot \delta\theta_0 \quad (19)$$

After carrier frequency estimating, measured pseudorange rate is obtained by

$$\dot{\rho} = -(1 + df_r)(f_{ca} - f_{IF})\lambda_{ca} \quad (20)$$

## 2.2. Navigation Processing

Navigation processing comprises two parts: Navigation update and navigation filter. The function of the navigation update is to predict the current navigation state according to the last navigation state. The function of the navigation filter is to estimate navigation errors according to pseudorange and pseudorange rate measurements.

### 2.2.1. Navigation Update

For high-dynamic scenarios, the position, velocity, and acceleration are used as the navigation state. Navigation update is based on the constant acceleration criterion, assuming constant acceleration within the update interval. The navigation update process is as follows:

$$\hat{L} = L^- + \frac{v_N^-}{R_M + h^-} T_{\text{nav}} \quad (21)$$

$$\hat{\lambda} = \lambda^- + \frac{v_E^-}{(R_N + h^-) \cos L^-} T_{\text{nav}} \quad (22)$$

$$\hat{h} = h^- - v_D^- T_{\text{nav}} \quad (23)$$

$$\hat{v}^n = v^{n-} + a^{n-} T_{\text{nav}} \quad (24)$$

$$\hat{a}^n = a^{n-} \quad (25)$$

$$\hat{b}_c = b_c^- + d_c^- T_{\text{nav}} \quad (26)$$

$$\hat{d}_c = d_c^- \quad (27)$$

Symbols in Equations (21)–(27):

$L, \lambda, h$  latitude, longitude, and altitude;

$v^n [v_E, v_N, v_D]$ , velocity vector in the geographic frame;

$a^n [a_E, a_N, a_D]$ , acceleration vector in the geographic frame;

$b_c$  equivalent range error of the receiver clock offset;

$d_c$  equivalent velocity error of the receiver clock drift;

$R_M, R_N$  curvature radiuses of the Earth in the meridian and prime vertical directions;

$T_{\text{nav}}$  navigation update period.

### 2.2.2. Navigation Filter

The errors of the navigation state are selected as state variables; the state vector is expressed as

$$\mathbf{X} = [\delta p^n, \delta v^n, \delta a^n, \delta b_c, \delta d_c]^T \quad (28)$$

The state equation is

$$\dot{\mathbf{X}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{X} \quad (29)$$

The measurement equation is

$$\mathbf{Z} = \begin{bmatrix} \delta \rho \\ \delta \dot{\rho} \end{bmatrix} = \begin{bmatrix} \rho' - \rho \\ \dot{\rho}' - \dot{\rho} \end{bmatrix} = \begin{bmatrix} e^n & \mathbf{0} & \mathbf{0} & -1 & \mathbf{0} \\ \mathbf{0} & e^n & \mathbf{0} & \mathbf{0} & -1 \end{bmatrix} \mathbf{X} \quad (30)$$

where  $e^n$  is the line-of-sight (LOS) unit vector from satellite to receiver. The satellite clock error and various errors in the signal propagation process [20] should be considered when calculating  $\rho'$  and  $\dot{\rho}'$ :

$$\rho' = \rho_r^s + c(dt_r - dt_s - dt_{rel} + dt_{GD} + dt_{Sagnac} + dt_{ion} + dt_{tro}) \quad (31)$$

$$\dot{\rho}' = \dot{\rho}_r^s + c(df_r - \delta f_s - \delta f_{rel} + \delta f_{Sagnac}) \quad (32)$$

Symbols in Equations (31) and (32):

- $\rho_r^s$  satellite-receiver geometric range;
- $dt_s$  satellite clock offset;
- $dt_{rel}$  relativistic correction;
- $dt_{GD}$  group delay;
- $dt_{Sagnac}$  Sagnac delay;
- $dt_{ion}$  ionosphere delay;
- $dt_{tro}$  tropospheric delay;
- $\dot{\rho}_r^s$  satellite-receiver relative velocity;
- $df_s$  satellite clock drift;
- $df_{rel}$  relativistic correction rate;
- $df_{Sagnac}$  Sagnac delay rate;
- $c$  speed of light.

To make the navigation filter more robust in the case of unstable measurement noise, a Huber-based Kalman filter is introduced [21,22]. The filtering process is given in Algorithm 1.

---

**Algorithm 1.** Huber-based Kalman filter for the navigation filter

---

- |        |                                                                                                                                                                                                                                                                                                                                                  |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | Calculate one-step prediction variance matrix:<br>$P_{k/k-1} = \Phi_{k/k-1} P_{k-1} \Phi_{k/k-1}^T + Q_{k-1}$ .                                                                                                                                                                                                                                  |
| Step 2 | Calculate initial state vector: $K_k = P_{k/k-1} H_k^T (H_k P_{k/k-1} H_k^T + R_k)^{-1}$ , $\hat{X}_k^0 = K_k Z_k$ .                                                                                                                                                                                                                             |
| Step 3 | Calculate residual vector: $\xi_X = P_{k/k-1}^{-1/2} (\hat{X}_k^j - \hat{X}_{k/k-1})$ , $\xi_Z = R_k^{-1/2} (H_k \hat{X}_k^j - Z_k)$ .<br>$j$ is the number of iterations, and its initial value is 0.<br>Calculate Huber weight matrix: $\Psi_{\xi, X} = \text{diag}[\mathcal{H}(\xi_X)]$ , $\Psi_{\xi, Z} = \text{diag}[\mathcal{H}(\xi_Z)]$ . |
| Step 4 | $\mathcal{H}$ is Huber function, $\mathcal{H}(\xi) = \begin{cases} 1, &  \xi  < \gamma \\ \frac{\gamma}{ \xi }, &  \xi  \geq \gamma \end{cases}$ , $\gamma$ is a parameter.                                                                                                                                                                      |
| Step 5 | Calculate $P'_{k/k-1}$ and $R'_k$ : $P'_{k/k-1} = P_{k/k-1}^{1/2} \Psi_{\xi, X}^{-1} (P_{k/k-1}^{1/2})^T$ , $R'_k = R_k^{1/2} \Psi_{\xi, Z}^{-1} (R_k^{1/2})^T$ .                                                                                                                                                                                |
| Step 6 | Calculate state vector: $K_k = P'_{k/k-1} H_k^T (H_k P'_{k/k-1} H_k^T + R'_k)^{-1}$ , $\hat{X}_k^{j+1} = K_k Z_k$ .                                                                                                                                                                                                                              |
| Step 7 | Check if $\ \hat{X}_k^{j+1} - \hat{X}_k^j\ $ is larger than a threshold. If yes, go back to Step 3.                                                                                                                                                                                                                                              |
| Step 8 | Calculate state vector error variance matrix: $P_k = (I - K_k H_k) P'_{k/k-1}$ .                                                                                                                                                                                                                                                                 |
- 

In Steps 5,  $P_{k/k-1}^{1/2}$  and  $R_k^{1/2}$  are lower triangular matrixes of the Cholesky factorization of  $P_{k/k-1}$  and  $R_k$ , respectively. In Step 3,  $P_{k/k-1}^{-1/2}$  and  $R_k^{-1/2}$  are inverse matrixes of  $P_{k/k-1}^{1/2}$  and  $R_k^{1/2}$ , respectively. The Huber function is a weight function of the residual, which reduces the weight of the measurement with large error.

Navigation state errors are obtained after each filter update and then used to correct the current navigation state. The process is as follows:

$$L^+ = \hat{L} - \frac{\delta p_N}{R_M + \hat{h}} \quad (33)$$

$$\lambda^+ = \hat{\lambda} - \frac{\delta p_E}{(R_N + \hat{h}) \cos \hat{L}} \quad (34)$$

$$h^+ = \hat{h} + \delta p_D \quad (35)$$

$$v^{n+} = \hat{v}^n - \delta v^n \quad (36)$$

$$a^{n+} = \hat{a}^n - \delta a^n \quad (37)$$

$$b_c^+ = \hat{b}_c + \delta b_c \quad (38)$$

$$d_c^+ = \hat{d}_c + \delta d_c \quad (39)$$

Process noise variance matrix  $Q$  is usually a diagonal matrix:

$$Q = \begin{bmatrix} Q_p & 0 & 0 & 0 & 0 \\ 0 & Q_v & 0 & 0 & 0 \\ 0 & 0 & Q_a & 0 & 0 \\ 0 & 0 & 0 & Q_{bc} & 0 \\ 0 & 0 & 0 & 0 & Q_{dc} \end{bmatrix} \quad (40)$$

$Q_p, Q_v,$  and  $Q_{bc}$  are unimportant parameters that can be set to zero.  $Q_a$  and  $Q_{dc}$  are the key parameters that affect the performance of the navigation filter and that need to be carefully considered. The magnitudes of  $Q_a$  and  $Q_{dc}$  depend on the maximal dynamic of the vehicle and the quality of the oscillator, respectively. Their values are generally set using a rule of thumb that the higher the vehicle's dynamic is, the larger the value of  $Q_a$  and the poorer the oscillator's quality is, the larger the value of  $Q_{dc}$ .

The measurement noise variance of each satellite is related to the intensity of the received signal. In our study, pseudorange noise variance is modeled as

$$\sigma_\rho^2 = \lambda_{co}^2 \frac{A_\rho}{n T_{coh} C/N_0} \quad (41)$$

where  $n$  is the number of average points in Equation (8),  $C/N_0$  is the carrier-to-noise ratio (CNR),  $A_\rho$  is a coefficient whose value is related to the autocorrelation function of the pseudo-code and the code phase discriminator model, which can be obtained by numerical simulation. The pseudorange rate noise variance is modeled as

$$\sigma_{\dot{\rho}}^2 = \lambda_{ca}^2 \frac{(A_\rho B_n)^3}{C/N_0} \quad (42)$$

where  $A_\rho$  is the coefficient for calculating the pseudorange rate noise variance, and it is also obtained by numerical simulation.

### 2.3. Weak Signal Channel Strategy

Although a third-order PLL can track high-dynamic signals, it is vulnerable. Because there are two integrators in the loop filter, the integral value is easy to diverge in the case of large phase discriminator noise. In practical tests, even a few seconds of a weak signal can cause third-order PLL loss of lock, but it cannot recover on its own ability when the signal becomes stronger. To ensure the continuity of signal tracking, it is necessary to design an additional reacquisition module when third-order PLL is used in a scalar receiver. In vector tracking, the fast recovery of third-order PLL is easy because of the superiority of the control strategy.

When the CNR of a channel is less than a preset threshold, the signal is judged as weak. In this case,  $f_{ca}$  and  $\dot{f}_{ca}$  in the pseudorange rate measurement module may be diverging. Therefore, the recovery of third-order PLL includes two aspects: Carrier frequency recovery and carrier frequency change rate recovery. The former is simple. Because  $f_{ca}^{NCO}$  is corrected by the receiver's velocity, it is supposed to be accurate when the navigation filter is working properly. So  $f_{ca}^{NCO}$  is assigned to  $f_{ca}$  after each navigation filter update. The latter requires

some additional calculations. First, the calculation formula of carrier frequency change rate is derived. The relative velocity between receiver and satellite is expressed as

$$v_{\text{los}} = (\mathbf{v}_s - \mathbf{v}_p) \cdot (\mathbf{r}_s - \mathbf{r}_p) \frac{1}{R} \quad (43)$$

where the subscript s denotes the satellite, subscript p denotes the receiver,  $\mathbf{r}$  denotes the position vector,  $\mathbf{v}$  denotes the velocity vector, and  $R = \|\mathbf{r}_s - \mathbf{r}_p\|$ . By taking the derivative of both sides of Equation (43) and considering  $\dot{R} = v_{\text{los}}$ , the relative acceleration between receiver and satellite can be obtained by

$$a_{\text{los}} = \frac{\mathbf{a}_{\text{ps}} \cdot \mathbf{r}_{\text{ps}}}{R} + \frac{\mathbf{v}_{\text{ps}} \cdot \mathbf{v}_{\text{ps}}}{R} - \frac{(\mathbf{v}_{\text{ps}} \cdot \mathbf{r}_{\text{ps}})^2}{R^3} \quad (44)$$

where  $\mathbf{a}$  denotes the acceleration vector,  $\mathbf{a}_{\text{ps}} = \mathbf{a}_s - \mathbf{a}_p$ ,  $\mathbf{v}_{\text{ps}} = \mathbf{v}_s - \mathbf{v}_p$ ,  $\mathbf{r}_{\text{ps}} = \mathbf{r}_s - \mathbf{r}_p$ . The relative acceleration is related to the velocity and acceleration of the satellite and receiver. The satellite's velocity and acceleration can be calculated by ephemeris [23,24]. The receiver's velocity is obtained by solving a velocity observation equation. The receiver's acceleration comes from the navigation state in Equation (25), but it is with respect to the geographic frame. Acceleration with respect to Earth-centered Earth-fixed (ECEF) frame  $\mathbf{a}_p$  is calculated with the Coriolis theorem:

$$\mathbf{a}_p = \mathbf{C}_n^e (\mathbf{a}^n + \boldsymbol{\omega}_{\text{en}}^n \times \mathbf{v}^n) \quad (45)$$

where  $\mathbf{C}_n^e$  is the transformation matrix of the geographic frame to the ECEF frame,  $\boldsymbol{\omega}_{\text{en}}^n$  is the angular velocity between the two frames. Then  $\dot{f}_{\text{ca}}$  can be calculated by  $a_{\text{los}}$  considering receiver clock drift:

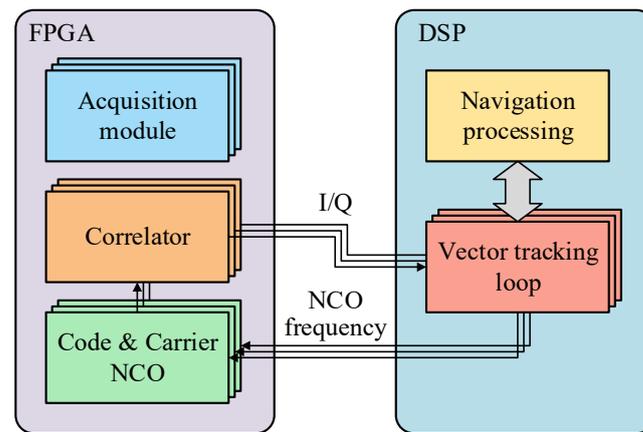
$$\dot{f}_{\text{ca}} = -\frac{a_{\text{los}}}{\lambda_{\text{ca}}(1 + df_r)} \quad (46)$$

Under the recovery action, the carrier phase tracking of weak signal degenerates into a first-order PLL, and the proportional term of carrier phase error  $K_1 \delta\theta$  is used to correct the carrier phase. For weak signals, because  $\delta\theta$  has a large error, phase locking is not guaranteed, and using  $K_1 \delta\theta$  correction is only an attempt. When the signal becomes stronger, due to the small error of  $f_{\text{ca}}$  and  $\dot{f}_{\text{ca}}$ , the carrier phase relocks, and the third-order PLL is restored. In this control mode, the recovery of an interrupted signal only needs a very short transition time, which is the phase locking time of a first-order PLL. The disadvantage that third-order PLL is an easy loss of lock is overcome.

## 2.4. Implementation of Hardware Platform

### 2.4.1. Structure of Hardware Platform

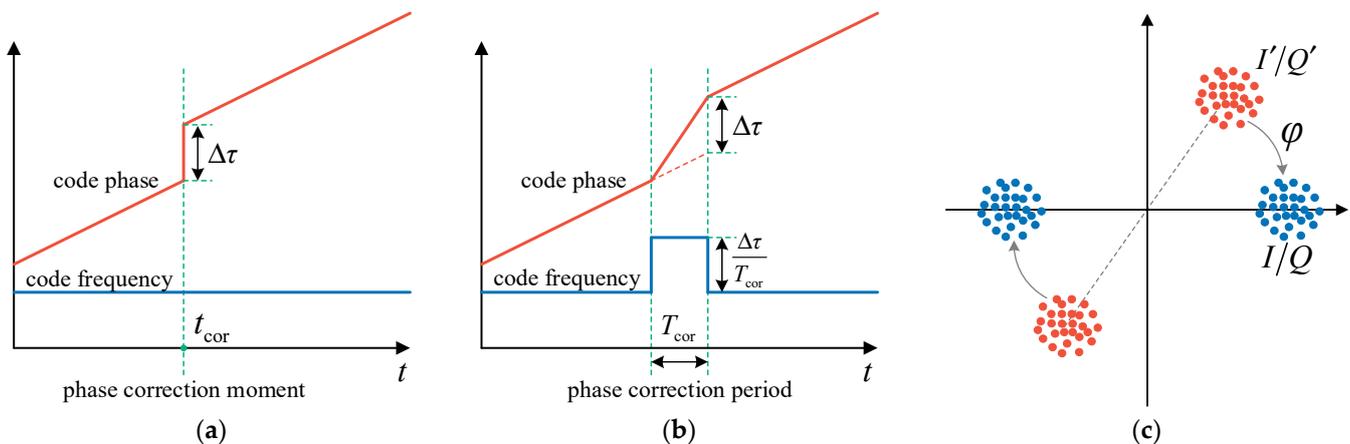
The vector tracking verification hardware platform used in this paper is an FPGA + DSP architecture. The functional diagram of the platform is shown in Figure 5. The FPGA is Intel's Cyclone V 5CEFA9F23I7, and the DSP is TI's TMS320C6748. The DSP scans the state of all channels every 0.505 ms. If a channel completes a correlation operation, the FPGA outputs the channel's I/Q correlation value. The vector tracking loop calculates the NCO frequency of each channel according to I/Q correlation values. Every 50 ms, all channels latch the current code phase, and FPGA generates an interrupt signal to trigger navigation processing in the DSP.



**Figure 5.** Functional diagram of vector tracking hardware platform.

#### 2.4.2. Code Phase Correction

Intuitively, code phase correction is to assign a desired phase or phase correction value to the code NCO at the correction moment, as shown in Figure 6a. However, the code phase of the code NCO cannot be directly controlled in practice. An indirect correction method is required. A phase correction period is set, typically 1 ms. In this period, code NCO frequency adds a phase correction frequency based on the normal value. At the end of the phase correction period, the code phase error is corrected, as shown in Figure 6b.



**Figure 6.** (a) Direct code phase correction; (b) indirect code phase correction; (c) carrier phase correction.

#### 2.4.3. Carrier Phase Correction

In practice, it is not necessary for the local replica carrier to be in phase with the actual carrier for signal tracking. Frequency matching is the only requirement. On the one hand, the incoherent code phase discriminator does not require carrier phase locking. On the other hand, the carrier phase error can be corrected after coherent integration. As shown in Figure 6c, when the phase error between the local and actual carrier is  $\varphi$ , the distribution of  $I/Q$  correlation values is orange dots.  $I/Q$  correlation values can be corrected to a zero phase by a rotation transformation of Equation (47). This is equivalent to phase locking.

$$\begin{cases} I = I' \cos \varphi + Q' \sin \varphi \\ Q = -I' \sin \varphi + Q' \cos \varphi \end{cases} \quad (47)$$

Thus, in receiver design, each channel has a variable used to record the current carrier phase error. The six  $I/Q$  outputs first perform Equation (47), and then perform the code

and carrier phase discriminator algorithm. Carrier phase correction can be realized by modifying the carrier phase error variable after each PLL update.

#### 2.4.4. Time Control

There are three basic time periods in VTL, which have been mentioned many times before. They are summarized in Table 1.

**Table 1.** Summary of three basic time periods in VTL.

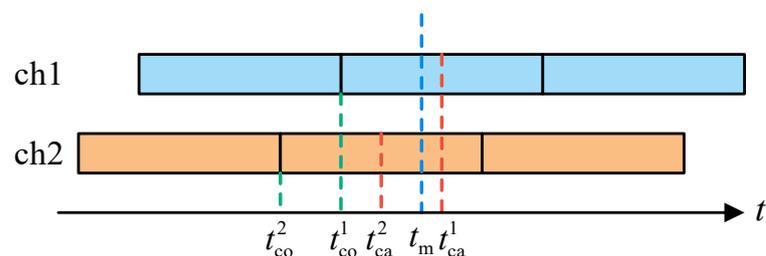
Symbol	Meaning	Description
$T_{\text{coh}}$	Coherent integration time	This depends on signal intensity. The weaker the signal is, the longer the coherent integration time.
$T_{\text{NCO}}$	Code and carrier NCO update period	For better tracking of high-dynamic signals, this period needs to be as small as possible, even if the coherent integration time is long. This is usually equal to the minimal integration time of the correlator, which was 1 ms on our hardware platform.
$T_{\text{nav}}$	Navigation update period	Theoretically, the faster the navigation update, the better. But fast navigation update has the problem of real-time computation. So, there is a trade-off to be made. Besides that, this period also depends on the hardware design. This was 50 ms on our hardware platform.

#### 2.4.5. Pseudorange Rate Measurement Synchronization

For the pseudorange measurement, all channels simultaneously latch the current code phases at the positioning moment, and they are synchronous. However, because the carrier frequency estimator update moment aligns with the end of coherent integration, all channels are asynchronous. Special processing is required to obtain synchronous pseudorange rate measurements. Figure 7 illustrates the process of pseudorange rate measurement.  $t_{\text{co}}^1$  and  $t_{\text{co}}^2$  are the end time of a pseudo-code period,  $t_{\text{ca}}^1$  and  $t_{\text{ca}}^2$  are the middle time of the next pseudo-code period, and  $t_m$  is the pseudorange rate measurement moment. In fact, the estimated carrier frequency at  $t_{\text{co}}$  is the carrier frequency at  $t_{\text{ca}}$ . Time difference  $t_m - t_{\text{ca}}$  needs to be considered to obtain the pseudorange rate measurements for all channels at  $t_m$ . The precise carrier frequency at  $t_m$  can be calculated by extrapolation using carrier frequency change rate:

$$f_{\text{ca}}^m = f_{\text{ca}} + \dot{f}_{\text{ca}}(t_m - t_{\text{ca}}) = f_{\text{ca}} + \dot{f}_{\text{ca}}[(t_m - t_{\text{co}}) - (t_{\text{ca}} - t_{\text{co}})] \quad (48)$$

where  $t_m - t_{\text{co}}$  is obtained by the code phase at  $t_m$ , and  $t_{\text{ca}} - t_{\text{co}}$  is equal to half of a pseudo-code period.



**Figure 7.** Diagram of pseudorange rate measurement process.

#### 2.4.6. Workflow of Vector Receiver

The workflow of a vector receiver can be divided into four stages. They are described in Table 2.

**Table 2.** Workflow of vector receiver.

Stage	Description
Before vector tracking	The receiver starts in scalar tracking mode and tracks satellite signals using conventional acquisition and tracking methods.
Enter vector tracking	When the number of satellites that are tracking and have ephemeris is greater than or equal to 4, the receiver enters vector tracking mode. The channels that are tracking signals are switched to use VTL.
Vector tracking running	Continuously acquire untracked satellites. When a new satellite is tracked, it is switched to use VTL.
Exit vector tracking	When the number of effective satellites (such as the CNR is larger than 23 dB·Hz) is less than 4 for a period of time, the navigation filter becomes untrusted, and the receiver exits vector tracking mode and returns to scalar tracking for a new working cycle.

### 3. Experiments and Results

GNSS signal simulator is efficient equipment to test a GNSS receiver, which can accurately and repeatedly generate GNSS signals received by a custom motion vehicle in a laboratory. In our study, a Spirent GSS7700 simulator was used to generate high-dynamic GPS L1 C/A signals for experiments. To create the desired motion scenario, a sequence containing a timestamp, position, and velocity was generated by self-developed trajectory generator software and then converted into a user motion file with a .umt extension. SimGEN software can process the file to simulate GPS signals. In our experiments, two high-dynamic scenarios were set up to verify the performance of VTL, an ultrahigh-dynamic scenario, and a normal high-dynamic scenario. Each scenario was tested with STL and VTL for comparison. To simply compare the performance of loop structures, the common parameters for STL and VTL were set the same. The detailed parameters are listed in Table 3.

**Table 3.** Receiver parameters configuration.

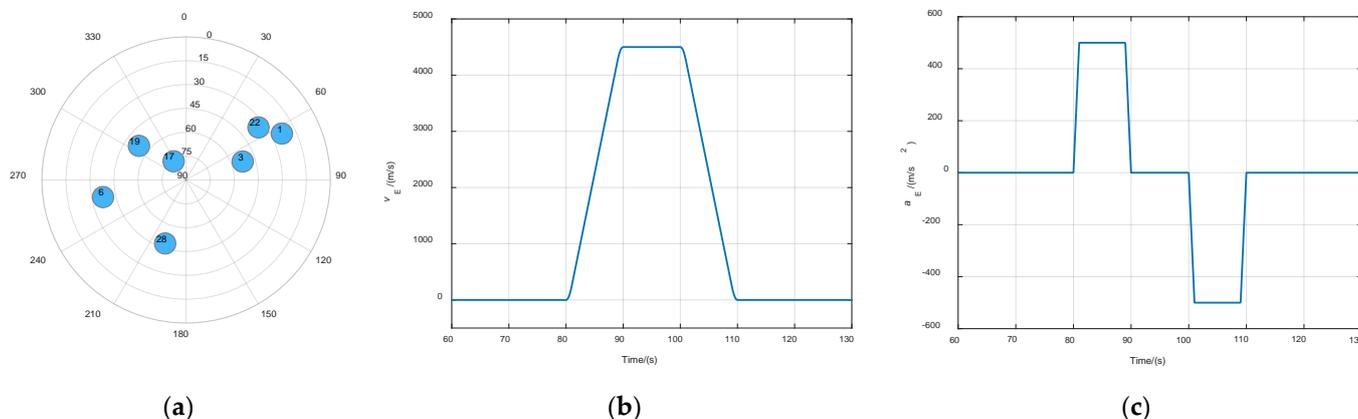
	Parameter	Value	Unit
STL	Bandwidth of 2nd-order DLL	2	Hz
	Bandwidth of 3rd-order PLL	18	Hz
	Coherent integration time *	1/5/20	ms
	Loss-of-lock threshold	18	dB·Hz
VTL	Bandwidth of 3rd-order carrier frequency estimator	18	Hz
	Coherent integration time *	1/5/20	ms
	Navigation update period	50	ms
	Loss-of-lock threshold	18	dB·Hz
	$Q_p$	0	$m^2$
	$Q_v$	0	$(m/s)^2$
	$Q_a$	100 <sup>2</sup>	$(m/s^2)^2$
$Q_{bc}$	0	$m^2$	
$Q_{dc}^\dagger$	0.3 <sup>2</sup>	$(m/s)^2$	

\* The coherent integration time is adjusted according to CNR, 1 ms for larger than 37 dB·Hz, 5 ms for between 37 and 30 dB·Hz, 20 ms for less than 30 dB·Hz in the experiment. <sup>†</sup> The oscillator used in the receiver is TCXO, whose stability is poor, so  $Q_{dc}$  is set to a relatively large value.

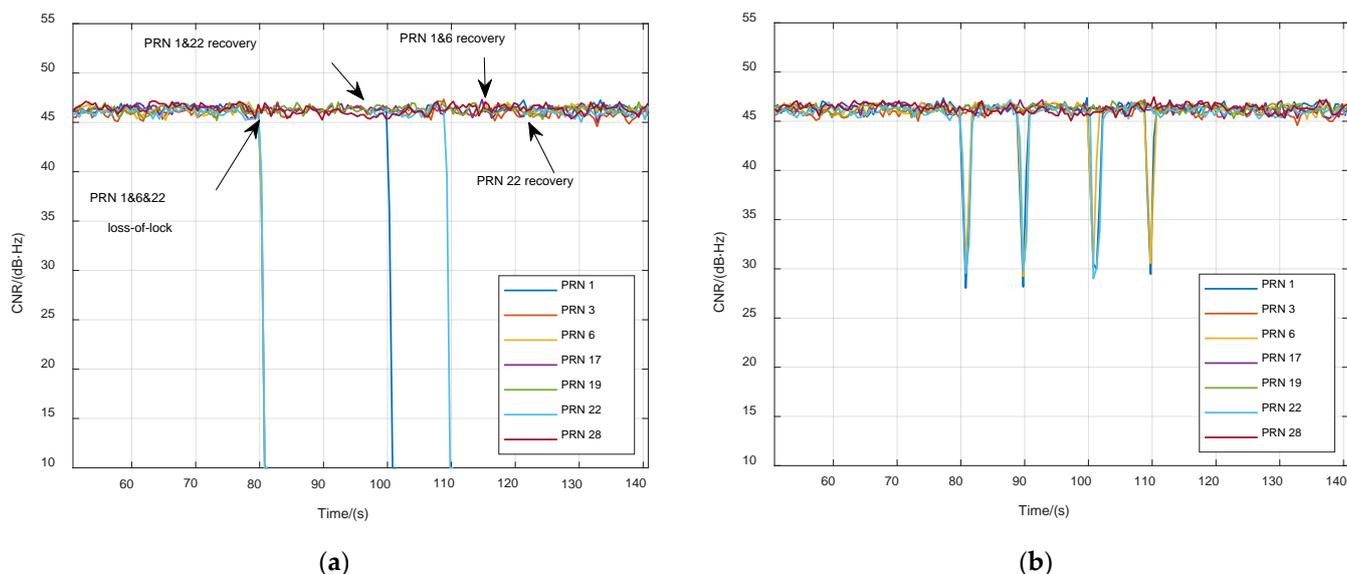
#### 3.1. Ultrahigh-Dynamic Scenario Experiment

The ultrahigh-dynamic scenario was an eastward accelerated trajectory in which the maximal acceleration was 50 g, the jerk was 50 g/s, and the maximal velocity was 4500 m/s. The satellite constellation, and the curves of velocity and acceleration of the trajectory are shown in Figure 8. Results of the two different tracking methods are shown in Figure 9. No.1, 6, and 22 satellites had large acceleration relative to the receiver, due to the lower elevating angle during the acceleration period. The carrier dynamic pressure of these signals exceeded the tracking threshold of the third-order PLL in STL, resulting in loss of lock at 80 s, 100 s, and 110 s, as shown in Figure 9a. Loss-of-lock signals need several seconds to recover. In VTL, the CNR of those three signals dropped only for two seconds at the moment of a sudden jerk because the four low-dynamic signals could maintain the

operation of the navigation filter, and the correct navigation state could provide accurate NCO control command for the high-dynamic channels to ensure effective tracking. Thus, the VTL showed better performance in high-dynamic signal tracking.



**Figure 8.** (a) Satellite constellation of experiment scenario; (b) curve of eastward velocity; (c) curve of eastward acceleration.



**Figure 9.** Carrier-to-noise ratio (CNR) of tracking satellites in ultrahigh-dynamic scenario. (a) Scalar tracking loop (STL); (b) vector tracking loop (VTL).

### 3.2. Normal High-Dynamic Scenario Experiment

The normal high-dynamic scenario was a figure-eight trajectory [25]. Trajectory length was 10,000 m, the average speed was 300 m/s, and altitude variation was 500 m. The curves of the 3D position, velocity, and acceleration of the trajectory are shown in Figure 10. This experiment contains three cases: Constant signal power, signal attenuation, and partial signal outage.

The purpose of the first case is to verify the improvement of VTL in navigation accuracy. The CNR of all signals was set to 46 dB·Hz. Results are shown in Figure 11. The position and velocity solution were both performed by single point mode in STL and VTL without other processes. Navigation root-mean-square (RMS) errors are listed in Table 4. The position error of VTL was obviously smaller than that of STL. The position error also reflected the tracking accuracy of the local replica code. Therefore, the code tracking accuracy of VTL was higher than that of STL. Theoretically, the same position accuracy could be obtained when the position output of STL passed the same navigation

filter as that of VTL, but the code phase of the NCO could not be changed in STL. The velocity errors of the two methods were almost the same, because the navigation filter's bandwidth was set to be relatively wide for tracking high-dynamic signals, resulting in no filtering effect on the velocity solution. Thus, the carrier frequency tracking accuracy of VTL cannot be improved unless the navigation filter's bandwidth is compressed, or an inertial measurement unit (IMU) is used.

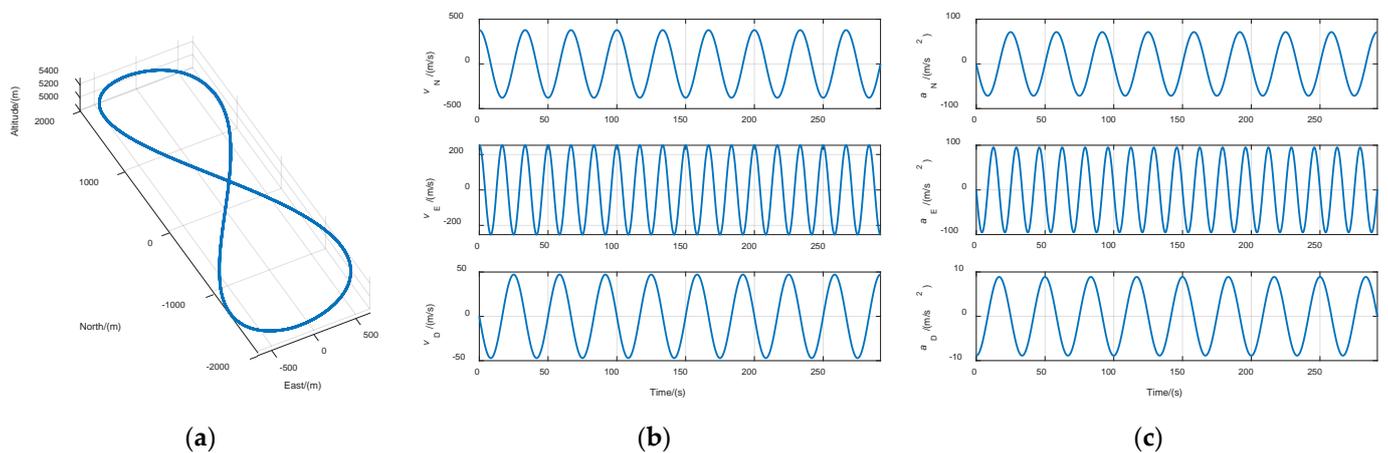


Figure 10. Simulated trajectory curves. (a) 3D position; (b) velocity; (c) acceleration.

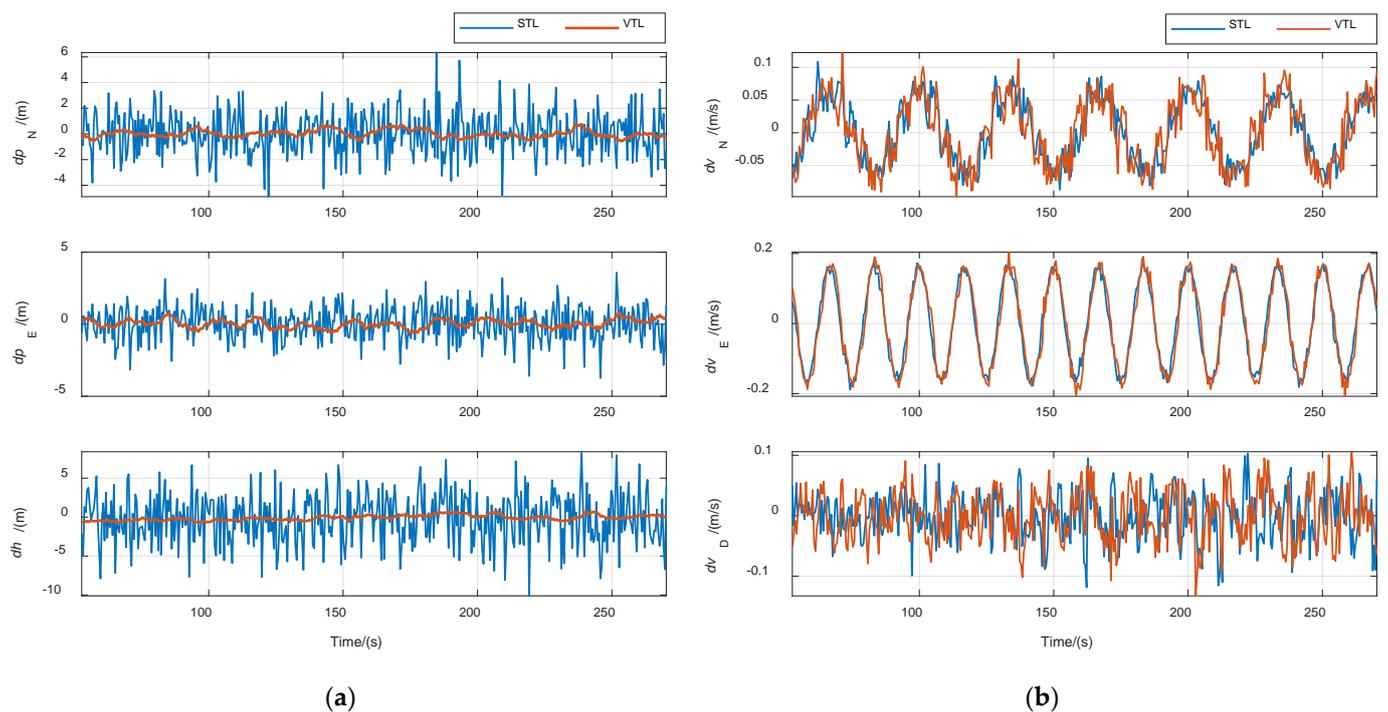


Figure 11. Comparison of navigation errors between STL and VTL in case of constant signal power. (a) Position error; (b) velocity error.

Table 4. Navigation root-mean-square errors statistics table.

	$dp_N/(m)$	$dp_E/(m)$	$dh/(m)$	$dv_N/(m/s)$	$dv_E/(m/s)$	$dv_D/(m/s)$
STL	1.68	1.17	3.21	0.047	0.115	0.039
VTL	0.28	0.29	0.34	0.051	0.120	0.042

The purpose of the second case was to verify the weak-signal-tracking performance of VTL. The initial CNR of all signals was set to 50 dB·Hz. Then the signal power was reduced at a rate of 1 dB/s from 90 s. From 200 s, the power increased at a rate of 1 dB/s until 50 dB·Hz. Results are shown in Figures 12 and 13. When the CNR dropped to 25 dB·Hz, STL channels lost the lock one after another, but all VTL channels kept tracking. During the weak signal period, the STL could not output navigation results. On the contrary, the VTL still provided relatively accurate navigation output, because it made use of the power of all channels. Thus, the tracking sensitivity of VTL was higher.

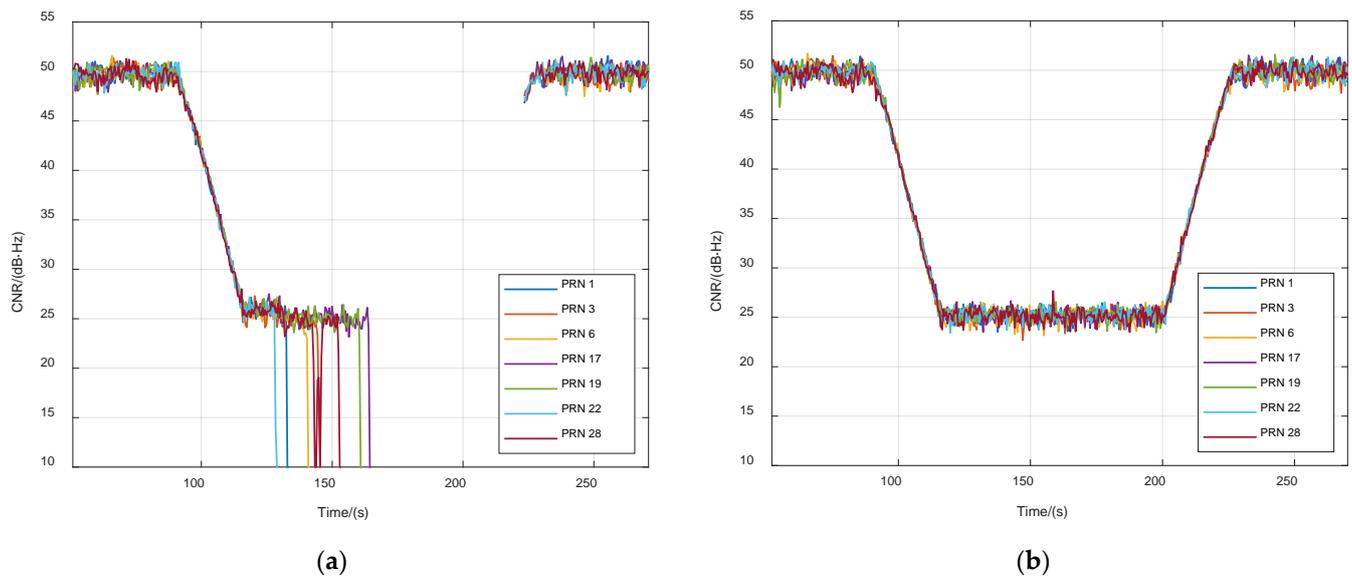


Figure 12. CNR of tracking satellites in case of signal attenuation. (a) STL; (b) VTL.

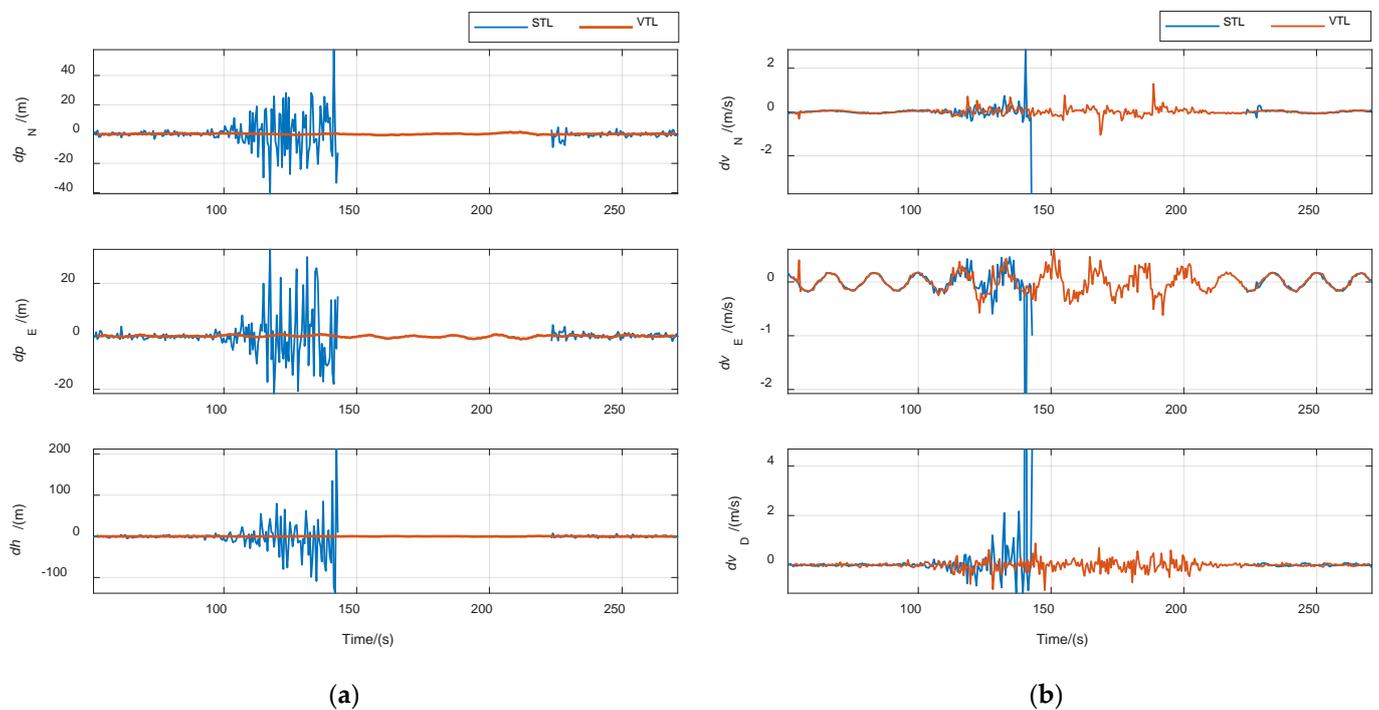
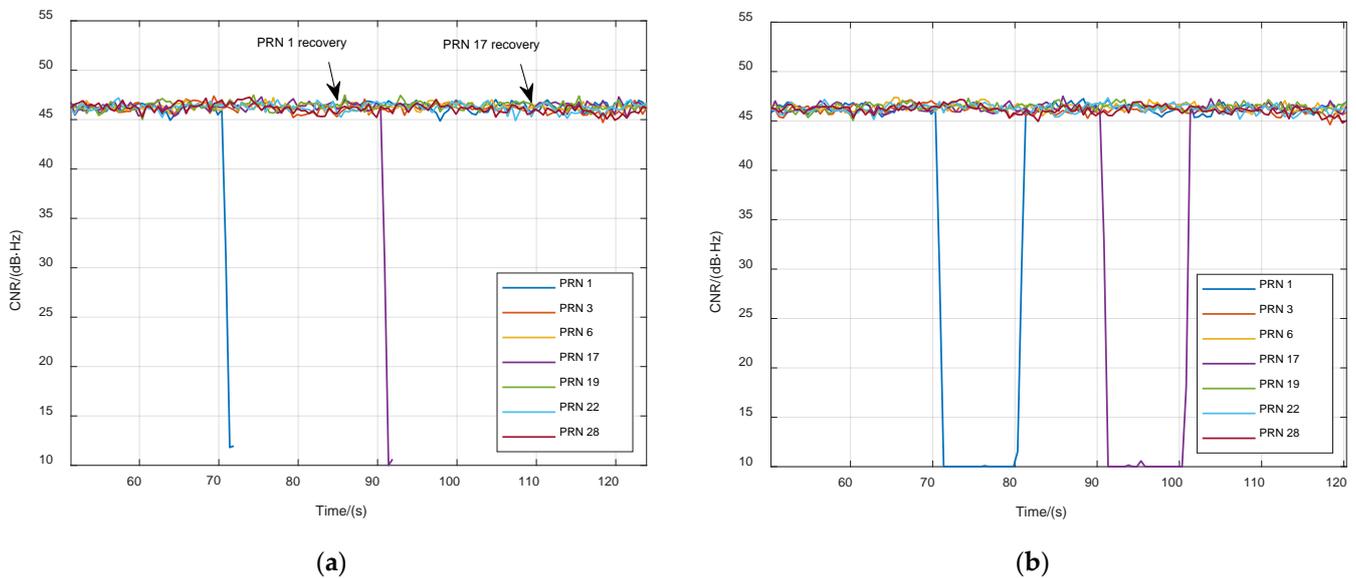


Figure 13. Comparison of navigation errors between STL and VTL in case of signal attenuation. (a) Position error; (b) velocity error.

The third case demonstrates the ability of VTL to quickly recover interrupted signals. The signals of PRN 1 and PRN 17 were closed for 10 s at 70 s and 90 s, respectively. Figure 14 shows the different results of STL and VTL. Signals were immediately relocked in VTL when signal power was recovered. In STL, the reacquisition process took longer.



**Figure 14.** CNR of tracking satellites in case of a partial signal outage. (a) STL; (b) VTL.

### 3.3. Calculation Time Statistics

Because the calculation of the code phase and carrier NCO frequency correction value, and the pseudorange and pseudorange rate measurement contain only a few simple arithmetic operations, the computation load of VTL is mainly on the navigation processing, in particular the navigation filter. To verify the real-time performance of VTL, the average calculation time of one instance of navigation processing was counted in cases of different numbers of tracking satellites. DSP frequency was 456 MHz. The statistical results are shown in Table 5. A cubic polynomial can be used to fit the number of satellites and calculation time. When the number of satellites processed by the navigation filter was 12, the proposed vector tracking method met the real-time requirement. Because NCO control takes almost no time, more satellites can be tracked in VTL; only a satellite selection algorithm is required to limit the number of satellites used by the navigation filter.

**Table 5.** The average calculation time of one instance of navigation processing.

Number of Satellites	Average Calculation Time/(ms)
4	1.3
6	2.6
8	4.7
10	8
12	12.8

## 4. Discussion

Although the study of VTL in this paper is based on GPS L1 C/A signal, which is a legacy signal using BPSK (binary phase shift keying) modulation, the VTL can also be used to track BOC (binary offset carrier) modulated signal, which is the new signal transmitted by modern GNSS satellites, such as GPS L1C signal, BDS B1C signal, and Galileo E1 signal. In BOC modulation, pseudo-code chips are multiplied by sine square wave subcarriers to realize the nature of split-spectrum [26]. In essence, the structure of the BOC signal is

still code and carrier. Therefore, the proposed VTL framework is applicable. Only the code generator, code phase discriminator, and carrier phase discriminator must be modified to adapt to the new signal.

In addition, the VTL has a potential in an antenna array-based GNSS receiver, which is used to mitigate RF interference. When using the minimum mean squared error (MMSE) algorithm to realize beamforming in the post-correlation technique, a reference signal is necessary [27]. The VTL can provide this reference signal more precisely.

At present, the receiver hardware platform still adopts the architecture of separated components, and DSP and FPGA chips are larger and consume more power. To further promote the practical application of vector tracking technology, the program will be transplanted to an SoC-based receiver platform, and an IMU will be introduced to assist.

## 5. Conclusions

Instead of using code phase and carrier frequency to directly control NCO, the vector tracking framework proposed in this paper calculates code phase and carrier frequency errors for incremental correction. This approach makes NCO control more flexible, so that VTL can be easily implemented based on existing receivers without much modification. A hardware receiver based on the proposed framework was realized on an FPGA + DSP platform with good real-time performance. Through effective information exchange between channels, the vector receiver had a stronger tracking ability of high-dynamic signals. In addition, the advantages of tracking sensitivity and bridging interrupted signals were verified.

**Author Contributions:** Conceptualization, R.M.; methodology, R.M.; software, T.L.; validation, T.L.; formal analysis, T.L.; investigation, R.M.; writing, T.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

**Acknowledgments:** Thanks go to the State Key Laboratory of Satellite Navigation System and Equipment Technology of China for providing the GPS simulator and receiver hardware platform support.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hinedi, S.; Statman, J.I. *High-Dynamic GPS Tracking Final Report*; Jet Propulsion Laboratory: Pasadena, CA, USA, 1988.
2. Scott, L.; Skone, S.; Raquet, J. Adaptive Antenna Arrays, Multi-GNSS Tropospheric Monitoring, and High-Dynamic Receivers. *Inside GNSS* **2006**, *4*, 20–25.
3. Abedi, M.; Jin, T. Improvement in tracking loop threshold of high dynamic GNSS receiver by installation of crystal oscillator on gyroscopic mounting. In Proceedings of the 2015 Joint Conference of the IEEE International Frequency Control Symposium and the European Frequency and Time Forum, Denver, CO, USA, 12–16 April 2015; pp. 522–527. [[CrossRef](#)]
4. Ma, L.; Shi, L.; Wang, Z. Performance analysis of a second order FLL assisted third order PLL for tracking doppler rates. *WSEAS Trans. Commun.* **2014**, *13*, 26–43.
5. Tu, Z.; Lu, T.; Chen, Q. A Novel Carrier Loop Based on Unscented Kalman Filter Methods for Tracking High Dynamic GPS Signals. In Proceedings of the 2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, China, 8–11 October 2018; pp. 1007–1012. [[CrossRef](#)]
6. Won, J.-H.; Pany, T.; Eissfeller, B. Iterative Maximum Likelihood Estimators for High-Dynamic GNSS Signal Tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 2875–2893. [[CrossRef](#)]
7. Xia, X.; Zhao, J.; Long, H.; Yang, G.; Sun, J.; Yang, W. Fractional Fourier transform-based unassisted tracking method for Global Navigation Satellite System signal carrier with high dynamics. *IET Radar Sonar Navig.* **2016**, *10*, 506–515. [[CrossRef](#)]
8. Luo, Y.; Yu, C.; Chen, S.; Li, J.; Ruan, H.; El-Sheimy, N. A Novel Doppler Rate Estimator Based on Fractional Fourier Transform for High-Dynamic GNSS Signal. *IEEE Access* **2019**, *7*, 29575–29596. [[CrossRef](#)]

9. Spilker, J.J., Jr.; Axelrad, P.; Parkinson, B.W.; Enge, P. *Global Positioning System: Theory and Applications, Volume I*; American Institute of Aeronautics and Astronautics: Washington DC, USA, 1996; pp. 290–305.
10. Petovello, M.G.; Lachapelle, G. Comparison of Vector-Based Software Receiver Implementations with Application 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), Fort Worth, TX, USA, 26–29 September 2006; pp. 1790–1799.
11. Tang, X.; Falco, G.; Falletti, E.; Presti, L.L. Performance comparison of a KF-based and a KF+VDFLL vector tracking-loop in case of GNSS partial outage and low-dynamic conditions. In Proceedings of the 2014 7th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), Noordwijk, The Netherlands, 3–5 December 2014; pp. 1–8. [[CrossRef](#)]
12. Babu, R.; Wang, J. Ultra-tight GPS/INS/PL integration: A system concept and performance analysis. *GPS Solut.* **2008**, *13*, 75–82. [[CrossRef](#)]
13. Hwang, D.H.; Lim, D.W.; Cho, S.L.; Lee, S.J. Unified approach to ultra-tightly-coupled GPS/INS integrated navigation system. *IEEE Aerosp. Electron. Syst. Mag.* **2011**, *26*, 30–38. [[CrossRef](#)]
14. Zhao, S.; Akos, D. An Open Source GPS/GNSS Vector Tracking Loop-Implementation, Filter Tuning, and Results. In Proceedings of the 2011 International Technical Meeting of the Institute of Navigation, San Diego, CA, USA, 24–26 January 2011; pp. 1293–1305.
15. Abbott, A.S.; Lillo, W.E. Global Positioning Systems and Inertial Measuring Unit Ultratight Coupling Method. U.S. Patent US6516021, 4 April 2003.
16. So, H.; Lee, T.; Jeon, S.; Kim, C.; Kee, C.; Kim, T.; Lee, S. Implementation of a vector-based tracking loop receiver in a pseudolite navigation system. *Sensors* **2010**, *10*, 6324–6346. [[CrossRef](#)] [[PubMed](#)]
17. Tu, Z.; Lou, Y.; Guo, W.; Song, W.; Wang, Y. Design and Validation of a Cascading Vector Tracking Loop in High Dynamic Environments. *Remote Sens.* **2021**, *13*, 2000. [[CrossRef](#)]
18. Watts, T.M. A GPS and GLONASS L1 Vector Tracking Software-Defined Receiver. Master’s Thesis, Auburn University, Auburn, AL, USA, 2019.
19. Kaplan, E.D.; Hegarty, C.J. *Understanding GPS: Principles and Applications*, 2nd ed.; Artech House: Norwood, UK, 2006; p. 180.
20. Teunissen, P.J.G.; Montenbruck, O. *Springer Handbook of Global Navigation Satellite Systems*; Springer: Cham, Switzerland, 2017; pp. 561–569.
21. Wang, X.; Cui, N.; Guo, J. Huber-based unscented filtering and its application to vision-based relative navigation. *IET Radar Sonar Navig.* **2010**, *4*, 134–141. [[CrossRef](#)]
22. Tseng, C.-H.; Lin, S.-F.; Jwo, D.-J. Robust Huber-Based Cubature Kalman Filter for GPS Navigation Processing. *J. Navig.* **2016**, *70*, 527–546. [[CrossRef](#)]
23. Zhang, J.; Zhang, K.; Grenfell, R.; Deakin, R. GPS Satellite Velocity and Acceleration Determination using the Broadcast Ephemeris. *J. Navig.* **2006**, *59*, 293–305. [[CrossRef](#)]
24. Thompson, B.F.; Lewis, S.W.; Brown, S.A.; Scott, T.M. Computing GPS satellite velocity and acceleration from the broadcast navigation message. *NAVIGATION J. Inst. Navig.* **2019**, *66*, 769–779. [[CrossRef](#)]
25. Grewal, M.S.; Weill, L.R.; Andrews, A.P. *Global Positioning Systems, Inertial Navigation, and Integration*; John Wiley: Hoboken, NJ, USA, 2007; p. 399.
26. Betz, J.W. Binary Offset Carrier Modulations for Radionavigation. *J. Inst. Navig.* **2001**, *48*, 227–246. [[CrossRef](#)]
27. Konovaltsev, A. Antenna Arrays for Robust GNSS in Challenging Environments. In Proceedings of the International Technical Symposium on Navigation and Timing (ITSNT), Toulouse, France, 17 November 2014.