GPS Solutions (2023) 27:127
https://doi.org/10.1007/s10291-023-01452-2

ORIGINAL ARTICLE

# GNSS position-aided delay-locked loops for accurate urban navigation

Tisheng Zhang[1] · Jinwei Shi[1] · Tao Lin[2] · Xin Feng[1] · Xiaoji Niu[1]

Received: 30 May 2022 / Accepted: 4 April 2023


## Abstract

In urban environments, when GNSS signals are weakened, blocked, and interfered with by multipath, the tracking accuracy and continuity of scalar delay-locked loops (SDLL) deteriorate seriously. Vector delay-locked loops (VDLL) can improve the tracking performance by estimating each channel's control information from positioning, but it requires reconstruction of the receiver baseband and has a heavy processing load. We propose a position-aided delay-locked loop (PA-DLL) with its numerically controlled oscillator (NCO) controlled by a local loop filter and position jointly. Compared with SDLL and VDLL, PA-DLL can be implemented with minor adjustments to the SDLL and brings the superiority of VDLL. NCO control by position is composed of an extended Kalman filter-based positioning algorithm and an aiding information estimation module. The performance of PA-DLL is fully tested in our software receiver i2xSNR and compared with SDLL and VDLL in urban environments. Compared to SDLL, the pseudorange and positioning accuracy of PA-DLL are improved by at least 20%. Test results show that PA-DLL performance significantly improved over SDLL and is comparable to VDLL with a high enough update rate. The processing load test illustrates that, compared to SDLL-based receiver, PA-DLL has only a little increase in processing load, while VDLL needs more than 10 times of computation load to ensure comparable performance.

**Keywords** Delay-locked loops · Vector tracking · Tracking loops · Global navigation satellite system (GNSS) receiver · GNSS pseudorange

### Abbreviations

| | |
|---|---|
| AIE | Aiding information estimation |
| BDS | BeiDou navigation satellite system |
| CDF | Cumulative distribution function |
| $C/N_0$ | Carrier-to-noise ratio |
| EKF | Extended Kalman filter |
| GNSS | Global navigation satellite system |
| GPS | Global positioning system |
| IF | Intermediate frequency |
| IMU | Inertial measurement unit |
| INS | Inertial navigation system |
| Is/Qs | In-phase (I) and quadrature (Q) signals |
| NCO | Numerically controlled oscillator |
| RAIM | Receiver autonomous integrity monitoring |
| PA-DLL | Position aided delay-locked loop |
| RTK | Real-time kinematic |
| SDLL | Scalar delay-locked loops |
| VDLL | Vector delay-locked loops |

✉ Xiaoji Niu
xjniu@whu.edu.cn

1 Integrated & Intelligent Navigation (i2Nav) Group, GNSS Research Center, Wuhan University, Wuhan 430072, China

2 Beijing Unistrong Science & Technology Co., Ltd., Beijing 100000, China

## Introduction

Autonomous vehicles, robots, and smart mobile terminals require positioning with high continuity, reliability and accuracy (Zhang et al. 2021b). GNSS can provide centimeter accurate positioning when real-time kinematic (RTK) or precise point positioning algorithm is used in open sky. However, the positioning deteriorates rapidly or even becomes unavailable in urban environments where GNSS signals are frequently blocked and interrupted (Li et al. 2018). INS is not affected by the surrounding environment. However, the measurement errors of INS with low-cost micro-electromechanical system (MEMS) inertial measurement unit (IMU) increase rapidly with time (Xu et al. 2018). Camera/LiDAR positioning is widely studied for unmanned systems. However, Camera/LiDAR are susceptible to sparse environmental features, dynamic objects and other factors (Gomez-Ojeda et al. 2019). Currently, multi-sensor fusion is the mainstream solution for continuous, reliable and accurate positioning

Published online: 09 May 2023

(Groves 2015). GNSS provides 24/7 absolute positioning and timing, which cannot be replaced by other sensors in the fusion solution (Parkinson 2014). Improving the availability of GNSS can reduce the dependence on the performance of other sensors, and the cost of the fusion solution.

To improve GNSS positioning accuracy in challenging environments, research on quality control of GNSS observations has been carried out (pseudorange and carrier phase). Zhang et al. (2019) used carrier-to-noise ratio ($C/N_0$) and elevation angle to evaluate the quality of GNSS observations. Ju et al. (2017) detected gross errors and cycle slips based on redundant observations. Zhang et al. (2021a, b) classified pseudorange based on machine learning. Du and Gao (2012) detected gross errors and cycle slips based on inertial navigation system (INS) and other sensors. Hsu et al. (2016) identified the path of observations based on 3D model. These researches mainly suppress GNSS positioning error by identifying those observations with poor quality and reducing their weight in positioning, which has no help to the accuracy and continuity of GNSS observations.

The GNSS signal tracking loops mainly determine the quality of observations. Almost all commercial receivers use the scalar tracking architecture, which is simple, low processing load, and easily implemented in hardware. The architecture of scalar tracking loops is shown in Fig. 1 (Kaplan and Hegarty 2017). Since the scalar loops track each satellite's signal independently, strong signal channels have no help to the weak signal channels. To improve the tracking performance and observation quality, vector tracking loops are proposed. As shown in Fig. 2, vector loops estimate code and carrier control information of all channels through position and velocity (Parkinson and Spilker 1996). Without base station information, the information estimated by positioning cannot meet the accuracy requirements of carrier phase tracking (Petovello et al. 2008). Therefore, vector loops mainly include VDLL and vector frequency locked loops. Since positioning is mainly based on pseudorange in the urban environment, we will discuss delay-locked loops.

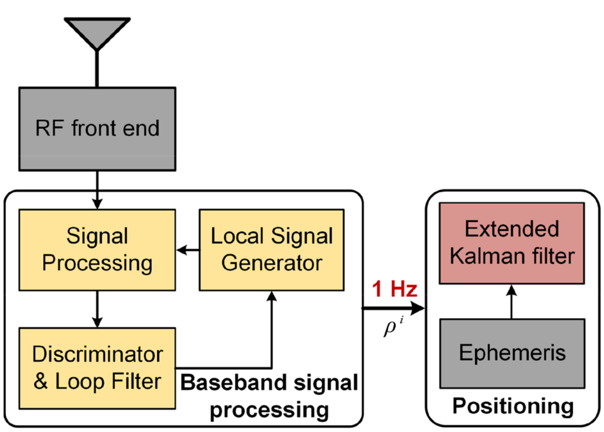



**Fig. 1** Scalar tracking loop architecture

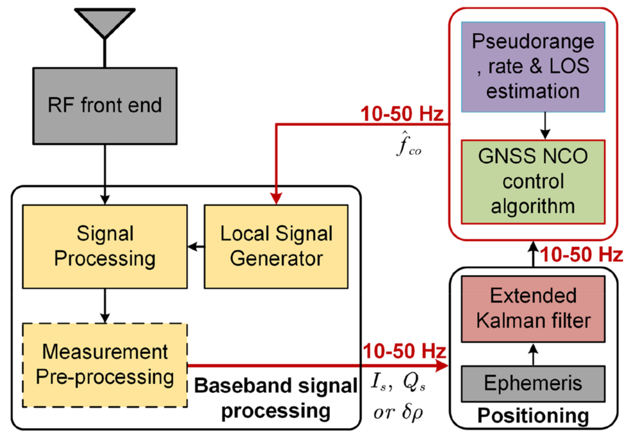



**Fig. 2** Vector tracking loop architecture

VDLL is generally divided into three categories: centralized coherent, cascaded coherent and cascaded non-coherent (Groves and Mather 2010). As shown in Fig. 2, the centralized coherent VDLL directly processes the in-phase (I) and quadrature (Q) signals (Is/Qs) in the Kalman filter (Gustafson et al. 2000) without pre-processing. Its processing load is high because of the large measurement vector and high update rate positioning (50 Hz). Therefore, the centralized coherent VDLL is rarely used in practical implementations. In the cascaded coherent VDLL, Is/Qs are pre-processed by channel filtering to obtain pseudorange/pseudorange increment first (Groves and Mather 2010), and the data rate is reduced to 10–20 Hz, which can reduce the processing load to a certain extent (Petovello et al. 2008). However, the cascaded coherent VDLL work is unsuited to applications in urban environments for its requirement of stable carrier phase tracking. In the cascaded non-coherent VDLL, Is/Qs are processed by a discriminator and filter first and then sent to the Kalman filter (Pany and Eissfeller 2006). Although the discriminator introduces nonlinear error, it does not need to track the carrier phase, which is suited for urban environments.

VDLL has been studied and assessed extensively. Through simulation analysis, Lashley et al. (2010) proved that VDLL has better sensitivity than SDLL. PLAN group from the University of Calgary implemented the VDLL on their software receiver platform and fully verified its superiority based on real GNSS signals in indoor environments (Petovello et al. 2008). The study of Pany et al. showed that VDLL helps track weak signals and increases the accuracy and continuity of pseudorange (Pany and Eissfeller 2006). Dardin et al. designed adaptive VDLL for reliable positioning (Dardin et al. 2013; Jiao et al. 2021). Hsu, L.T. et al. studied multipath detection using VDLL in urban environments (Jiang et al. 2021). The performance tests of VDLL on the vehicle in urban environments by Ren and Petovello (2017) showed that VDLL could improve the accuracy of GNSS signal tracking and positioning. Therefore, VDLL can improve signal tracking performance in challenging environments by sharing information between different channels through positioning.

Although the performance of VDLL is obviously superior, it is rarely used in commercial receivers. The architecture of VDLL is too different from the SDLL to be implemented with minor adjustments of the SDLL in commercial receivers. There are some drawbacks of VDLL relative to SDLL in implementation. First of all, the architecture of VDLL is more complicated. It needs the baseband signal processing and positioning to work together and initializes via SDLL-based positioning. Besides, the processing load of VDLL is heavy. A high update rate positioning is required to ensure NCO updating in time. Finally, VDLL is subject to positioning reliability. When positioning deteriorates or even fails, all satellites may lose lock. (Lashley et al. 2009; Groves and Mather 2010).

We propose PA-DLL to overcome the shortcomings of both SDLL and VDLL. PA-DLL can be implemented with minor adjustments to the SDLL. Our main contributions to our research include:

1. Position aided delay-locked loop is proposed, which improves code phase tracking performance and avoids the drawbacks of VDLL in implementation.
2. Key units of the PA-DLL are described, including an EKF-based positioning algorithm and an aiding information estimation module.
3. Performance and processing load of PA-DLL, SDLL, and VDLL with different positioning update rates are fully compared by vehicle tests in urban environments.

## Methodology

This section focuses on the design of PA-DLL. The architecture of the position aided delay-locked loop is proposed and compared with SDLL and VDLL. Then, the positioning algorithm based on EKF is described. Finally, the aiding information estimation module is discussed.

### PA-DLL architecture

The architecture of the PA-DLL is shown in Fig. 3, which mainly includes three components: baseband signal processing, positioning, and aiding information estimation module (AIE). It is clear from Fig. 3 that PA-DLL adopts code phase error obtained from AIE to aid the DLL. The left part is baseband signal processing, whose signal processing

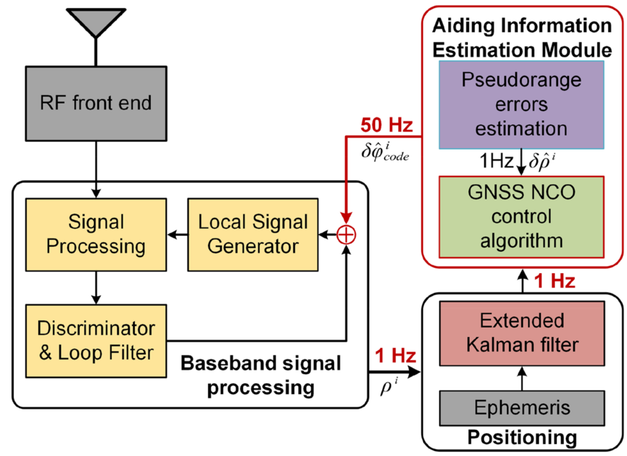



**Fig. 3** Position-aided delay-locked loop architecture

components and observation interfaces are exactly the same as the conventional SDLL. The positioning module can directly utilize the SDLL-based positioning algorithm in the lower right part compatible. The AIE module in the upper right includes pseudorange error estimation and code NCO control information generation. The input of the AIE includes pseudorange observations, position and velocity of the satellite and receiver, clock errors of the receiver (clock bias and drift) and the correction information of pseudorange. The input rate is 1 Hz. The output of AIE is the estimation of code phase error, which is sent to baseband signal processing to assist code NCO control. Under normal conditions, the update rate of DLL is 50 Hz. Accordingly, the output update rate of AIE is required to be 50 Hz. To obtain 50 Hz output, we adopt one linear model to predict code phase error in one second when generating aiding information. PA-DLL supports three code NCO control modes: loop filter output, external aiding information, or both of them. The control modes of code NCO can be selected manually or adaptively adjusted. Therefore, the PA-DLL can be implemented by only adding the AIE module to the SDLL, without changing any existing components. Compared with the VDLL, PA-DLL has multiple advantages in implementation, as shown follows:

1. PA-DLL can be implemented without restructuring the SDLL. In VDLL, the code NCO is directly controlled by information estimated from the position with the control information from the loop filter output cutoff. In other words, to implement VDLL, the SDLL needs to be removed. PA-DLL joints the filter output and the position estimated information to control the code NCO, which does not need to cut off the loop filter.
2. PA-DLL does not need to change the interface between the baseband and positioning. The input of the PA-

DLL's positioning unit is pseudorange $\rho^i$, which is the same as common receivers. While the Is/Qs or pseudorange increment $\delta\rho^i$ is sent to the positioning unit in the VDLL, which needs to adjust the interface between the baseband and the positioning. In addition, SDLL usually has an interface for carrier assistance, which can also be deemed as a source of aiding information to support code phase tracking.

3. PA-DLL can overcome the heavy processing load and control the delay of VDLL. To control the code NCO accurately, VDLL generally carries out positioning with an update rate of no less than 10 Hz to estimate the code phase, significantly increasing the processing load and NCO control delay. Thanks to the high update rate loop filter output, PA-DLL supports code phase error estimation with a much lower positioning update rate (such as 1 Hz), which overcomes the heavy processing load and control delay of VDLL.
4. PA-DLL can improve the integrity of NCO control information. The tracking performance of the VDLL is restricted by positioning. When the positioning deteriorates or even fails, all satellites can only fall back to reacquisition. However, PA-DLL can degenerate into SDLL and keep tracking those satellites with strong signal strength. In addition, the control information from the high update rate loop filter can eliminate errors caused by low update rate aiding information from positioning.

## Extended Kalman filter-based positioning

To suppress the position gross error, the EKF-based positioning is implemented in PA-DLL. It estimates the errors in the position, velocity, and clock for low dynamic carriers. The state vector can be expressed as (Hsu et al. 2015):

$$\boldsymbol{\delta}\mathbf{x} = \left[\delta r_{\mathrm{pos\_x}}\ \delta r_{\mathrm{pos\_y}}\ \delta r_{\mathrm{pos\_z}}\ \delta r_{\mathrm{vel\_x}}\ \delta r_{\mathrm{vel\_y}}\ \delta r_{\mathrm{vel\_z}}\ \delta\rho_\tau\ \delta\dot{\rho}_\tau\right]^{\mathrm{T}} \tag{1}$$

where $\boldsymbol{\delta r}_{\mathrm{pos}}$, $\boldsymbol{\delta r}_{\mathrm{vel}}$, $\delta\rho_\tau$, and $\delta\dot{\rho}_\tau$ are the position error, velocity error, receiver clock bias, and drift, respectively. The clock bias and drift are expressed as a range and range rate, respectively. The state transition matrix is expressed as:

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_{3\times3} & T_s\mathbf{I}_{3\times3} & \mathbf{0}_{3\times2} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times2} \\ \mathbf{0}_{2\times3} & \mathbf{0}_{2\times3} & \mathbf{A_c} \end{bmatrix}, \quad \mathbf{A_c} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \tag{2}$$

where $T_s$ is the EKF update interval, $\mathbf{A_c}$ is the state transition matrix of the clock.

The system (process) noise covariance matrix can be divided into dynamic receiver noise and receiver clock noise (Parkinson and Spilker 1996):

$$\mathbf{Q} = \begin{bmatrix} S_v\frac{T_s^3}{3}\cdot\mathbf{I}_{3\times3} & S_v\frac{T_s^2}{2}\cdot\mathbf{I}_{3\times3} & \mathbf{0}_{3\times2} \\ S_v\frac{T_s^2}{2}\cdot\mathbf{I}_{3\times3} & S_vT_s\cdot\mathbf{I}_{3\times3} & \mathbf{0}_{3\times2} \\ \mathbf{0}_{2\times3} & \mathbf{0}_{2\times3} & \mathbf{Q}_c \end{bmatrix}, \quad \mathbf{Q}_c = \begin{bmatrix} Q_{c11} & Q_{c12} \\ Q_{c12} & Q_{c22} \end{bmatrix} \tag{3}$$

where

$$Q_{c11} = \frac{h_0}{2T_s} + 2h_{-1}T_s^2 + \frac{2}{3}\pi^2h_{-2}T_s^3 \tag{4}$$

$$Q_{c12} = 2h_{-1}T_s + \pi^2h_{-2}T_s^2 \tag{5}$$

$$Q_{c22} = \frac{h_0}{2T_s} + 2h_{-1} + \frac{8}{3}\pi^2h_{-2}T_s \tag{6}$$

where $S_v$ is the receiver velocity noise power spectral density (PSD), $h_0$, $h_{-1}$ and $h_{-2}$ are the parameters of receiver clock noise.

Measurement innovations input to the EKF includes the delta pseudorange, which is the difference between the measured and predicted pseudorange, and the delta Doppler, which is the difference between the measured and predicted Doppler. The measurement innovation vector is given as:

$$\boldsymbol{\delta}\mathbf{z} = \left[\delta\rho^1\ \delta\rho^2\ \cdots\ \delta\rho^i\ \delta f_{\mathrm{d}}^1\ \delta f_{\mathrm{d}}^2\ \cdots\ \delta f_{\mathrm{d}}^i\right]^{\mathrm{T}} \tag{7}$$

where $i$ is the number of satellites, $\delta\rho^i$ and $\delta f_{\mathrm{d}}^i$ are, the delta pseudorange and delta Doppler, respectively. The measurement matrix, which comprises the partial derivatives of the measurements with respect to the states, is:

$$\mathbf{H} = \begin{bmatrix} \mathbf{U} & \mathbf{0}_{i\times3} & 1_{i\times1} & \mathbf{0}_{i\times1} \\ \mathbf{0}_{i\times3} & \mathbf{U}/\lambda & \mathbf{0}_{i\times1} & 1_{i\times1}/\lambda \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_x^1 & u_y^1 & u_z^1 \\ u_x^2 & u_y^2 & u_z^2 \\ \vdots & \vdots & \vdots \\ u_x^i & u_y^i & u_z^i \end{bmatrix} \tag{8}$$

where $\lambda$ is carrier wavelength, and $\boldsymbol{u}$ is the line of sight (LOS) unit vector from the satellite to the receiver.

The measurement noise covariance matrix, $\mathbf{R}$, is determined adaptively by the elevation angle of the satellite, which is reliable in open sky by not in urban environments. Therefore, one receiver autonomous integrity monitoring (RAIM) algorithm is employed in the EKF to monitor the integrity of satellite signals and identify faulty satellites in time. The RAIM algorithm based on the innovation monitoring method is expressed as follows (Wang et al. 2019; Liu et al. 2019):

$$\begin{cases} \mathbf{q}_{k,i} \le T_{\mathrm{d}}^2 & \text{measurement is valid} \\ \mathbf{q}_{k,i} > T_{\mathrm{d}}^2 & \text{measurement is invalid} \end{cases} \tag{9}$$

$$\mathbf{q}_{k,i} = \mathbf{V}_{k,i}^{T}\mathbf{W}_{k,i}^{-1}\mathbf{V}_{k,i} \tag{10}$$

$$\mathbf{V}_{k,i} = \boldsymbol{\delta}\mathbf{z}_{k,i} - \mathbf{H}_{k,i}\boldsymbol{\delta}\mathbf{x}_{k|k-1} \tag{11}$$

$$\mathbf{W}_{k,i} = \mathbf{H}_{k,i}\mathbf{P}_{k|k-1}\mathbf{H}_{k,i}^{T} + \mathbf{R}_{\boldsymbol{k,i}} \tag{12}$$

where $\mathbf{q}_{k,i}$ is the statistic for satellite $i$ result at epoch k, and $T_{\mathrm{d}}$ is the detection threshold. $T_{\mathrm{d}}$ directly affects the accuracy of detection. So $T_{\mathrm{d}}$ should be set to an appropriate value. $\mathbf{V}_{k,i}$ and $\mathbf{W}_{k,i}$ are the innovations and covariance matrix corresponding to the innovations, $\mathbf{P}_{k|k-1}$ is the predicted covariance matrix corresponding to the states at epoch $k$. It is clear from equation (12) that the accurate $\mathbf{P}_{k|k-1}$ and $\mathbf{R}_{\boldsymbol{k,i}}$ will improve the reliability of RAIM. In general, the test statistic $\mathbf{q}_{k,i}$ should have chi-squared distribution with one degree of freedom:

$$\mathbf{q}_{k,i} \sim \chi_1^2 \tag{13}$$

When faulty signals that may cause positioning error are detected, the EKF-based positioning is recomputed to obtain a reliable position in the urban environment, which is helpful to the integrity of NCO control information.

## Code phase error estimation model

Taking global positioning system (GPS) L1 signals as an example, considering main error factors, the pseudorange observation equation can be expressed as (Teunissen and Kleusberg 1998):

$$\rho^i = r^i + \delta\rho_\tau + \delta\rho_c^i + \varepsilon_\rho^i \tag{14}$$

where $i$ is the satellite number, $\rho^i$ is the measured pseudorange, $r^i$ is the true distance between satellite and receiver, $\delta\rho_\tau$ is the clock error of the receiver, $\delta\rho_c^i$ includes errors caused by the clock of the satellite and propagation path, and $\varepsilon_\rho^i$ is the observation errors of the receiver. $r^i$ is expressed as:

$$r^i = \left\| \boldsymbol{s}_{\mathrm{pos}}^i - \boldsymbol{r}_{\mathrm{pos}} \right\| \tag{15}$$

where $\boldsymbol{s}_{\mathrm{pos}}^i$ is the satellite position, and $\boldsymbol{r}_{\mathrm{pos}}$ is the receiver position. $\delta\rho_c^i$ is:

$$\delta\rho_c^i = \delta\rho_I^i + \delta\rho_T^i - \delta\rho_s^i + \delta\rho_{er}^i + \delta\rho_{gd}^i \tag{16}$$

where $\delta\rho_I^i$ is the ionospheric delay, $\delta\rho_T^i$ is the tropospheric delay, $\delta\rho_{\mathrm{s}}^i$ is the clock bias of satellite, $\delta\rho_{er}^i$ is the rotation delay of the earth, and $\delta\rho_{\mathrm{gd}}^i$ is the group delay.

Referring to (14), the estimated pseudorange can be derived as:

$$\hat{\rho}^i = \hat{r}^i + \delta\hat{\rho}_\tau + \delta\hat{\rho}_c^i \tag{17}$$

where $\hat{\rho}^i$ is the estimated pseudorange, $\hat{r}^i$ is the estimated distance between the satellite and receiver, $\hat{\rho}_\tau$ is the estimated clock error of the receiver, $\hat{\rho}_c^i$ is the estimated errors caused by the clock of the satellite and propagation path. $\hat{r}^i$ can be written as:

$$\hat{r}^i = \left\| \hat{\boldsymbol{s}}_{\mathrm{pos}}^i - \hat{\boldsymbol{r}}_{\mathrm{pos}} \right\| \tag{18}$$

where $\hat{\boldsymbol{s}}_{\mathrm{pos}}^i$ is the estimated satellite position calculated by broadcast ephemeris. In general, the position error of satellite is less than 1 m. $\hat{\boldsymbol{r}}_{\mathrm{pos}}$ is the estimated receiver position provided by the EKF. $\delta\hat{\rho}_c^i$ is expressed as:

$$\delta\hat{\rho}_{\mathrm{c}}^i = \delta\hat{\rho}_{\mathrm{I}}^i + \delta\hat{\rho}_{\mathrm{T}}^i - \delta\hat{\rho}_{\mathrm{s}}^i + \delta\hat{\rho}_{\mathrm{er}}^i + \delta\hat{\rho}_{\mathrm{gd}}^i \tag{19}$$

where $\delta\hat{\rho}_{\mathrm{I}}^i$ is the estimated ionospheric delay, $\delta\hat{\rho}_{\mathrm{T}}^i$ is the estimated tropospheric delay, $\delta\hat{\rho}_{\mathrm{s}}^i$ is the estimated clock error of satellite, $\delta\hat{\rho}_{\mathrm{er}}^i$ is the estimated rotation delay of earth, and $\delta\hat{\rho}_{\mathrm{gd}}^i$ is the estimated group delay. $\delta\hat{\rho}_c^i$ with meter level accuracy can be estimated by broadcast ephemeris and atmospheric models. $\delta\hat{\rho}_c^i$ will be centimeter-level accurate when precise ephemeris is adopted. The estimated value of pseudorange error $\delta\hat{\rho}^i$ is obtained by evaluating the difference between the measured pseudorange $\rho^i$ and the estimated pseudorange $\hat{\rho}^i$:

$$\delta\hat{\rho}^i = \rho^i - \hat{\rho}^i = r^i - \hat{r}^i + \delta\rho_{\mathrm{c}}^i - \delta\hat{\rho}_{\mathrm{c}}^i + \delta\rho_\tau - \delta\hat{\rho}_\tau + \varepsilon_\rho^i \tag{20}$$

Unit conversion is required when using $\delta\hat{\rho}^i$ to aid DLL. The relationship between code phase error $\delta\hat{\varphi}_{\mathrm{code}}^i$ and $\delta\hat{\rho}^i$ is derived as:

$$\delta\hat{\varphi}_{\mathrm{code}}^i = \frac{\delta\hat{\rho}^i}{\lambda_{\mathrm{code}}} \tag{21}$$

where $\lambda_{\mathrm{code}}$ is the code wavelength. In urban environments, the error of measured pseudorange caused by multipath, etc., may be tens or even hundreds of meters. The impact of $\delta\hat{\rho}_c^i$ and $\hat{\boldsymbol{s}}_{\mathrm{pos}}^i$ on $\delta\hat{\varphi}_{\mathrm{code}}^i$ is at the meter level. The receiver clock error $\delta\hat{\rho}_\tau$ obtained in positioning is processed by a Kalman filter to suppress gross errors.

Besides the accuracy of the estimated code phase errors, the update rate and delay of the code NCO control information also affect the tracking performance. Therefore, we will discuss the generation of code NCO control information in the PA-DLL.

## Code NCO control information generation

Since the processing load of the filter and discriminator is relatively light, SDLL can work with a high update rate and low delay. However, the NCO control information severely deteriorates in complex situations such as multipath. VDLL estimates the NCO control information directly by position and improves the accuracy of NCO control information in challenging environments. However, the update rate and delay of VDLL's NCO control information are affected by the heavy processing load of positioning.

Figure 3 shows that PA-DLL joints the output of the loop filter and the estimated code phase error to control the NCO. When the update rate of measured pseudorange and positioning is 1 Hz, PA-DLL only obtains the estimated code phase errors with 1 Hz. To control the code NCO, generating the aiding information with the same update rate to the loop filter output is necessary. So we adopt a linear model to predict code phase error in one second. The aiding information with a high update rate is expressed as follows:

$$\delta\hat{\varphi}^i_{\mathrm{code\_}k+n} = \frac{\delta\hat{\rho}^i_k}{\lambda_{\mathrm{code}}}\left(1-\frac{n}{N}\right) \quad n = 0, 1, \cdots, N-1 \tag{22}$$

where N is the number of filter updates in one second, $\delta\hat{\varphi}^i_{\mathrm{code\_}k+n}$ is the n-th estimated code phase error after the k-th positioning.

When the signal is reflected or momentarily blocked, a step error of the code phase happens, which will be estimated quickly and accurately by AIE. The loop filter output error will gradually reduce as the estimated code phase error is sent to the NCO. In this case, the value of the aiding information should be gradually reduced within 1 s. When the signal is weakened, the code discriminator cannot reflect the code phase error, and the loop filter output is not reliable. Therefore, the value of the aiding information should keep the same within 1 s. So, according to the $C/N_0$, equation (22) is derived as:

$$\delta\hat{\varphi}^i_{\mathrm{code\_}k+n} = \begin{cases} \frac{\delta\hat{\rho}^i_k}{\lambda_{\mathrm{code}}}\left(1-\frac{n}{N}\right) & C/N_0 \geq 32\ \mathrm{dB\text{-}Hz}, \quad n = 0, 1 \cdots, N-1 \\ \frac{\delta\hat{\rho}^i_k}{\lambda_{\mathrm{code}}} & C/N_0 \leq 32\ \mathrm{dB\text{-}Hz} \end{cases} \tag{23}$$

Among the factors that affect the accuracy of the estimated code phase error, the satellite position and velocity,


**Fig. 4** Test devices on the vehicle

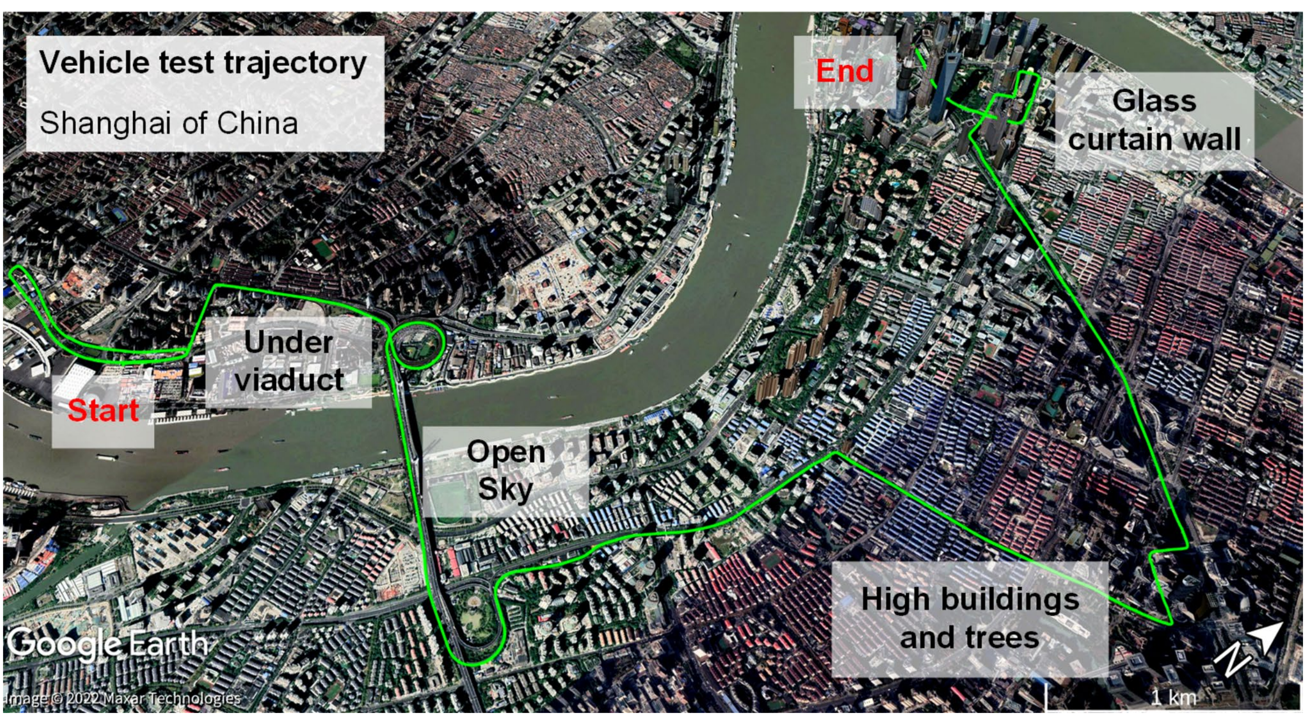



**Fig. 5** Test situations and reference trajectory (green lines, Google Earth)


**Fig. 6** Typical testing situations: under viaduct (top left), open sky (top right), high buildings and trees (bottom left), glass curtain wall (bottom right)

ionosphere and troposphere can be negligible. If the receiver positioning deteriorates, the accuracy of the estimated information will be affected. To ensure the availability and accuracy of position, the EKF-based positioning with RAIM is employed, and the impact of the estimated receiver clock error can be mitigated by using a clock model.

## Experimental setup

Experiments are carried out to assess the performance of the proposed PA-DLL in urban environments. Test devices on the vehicle are shown in Fig. 4. In this test, a centimeter-level POS system, Leador-A15, is served as the reference system. A GNSS recording and playback system (GSS6450) was employed to record intermediate frequency (IF) data of GPS L1 and BeiDou navigation satellite system (BDS) B1I. Two devices are connected to the antenna through a power splitter which is fixed on the top of the vehicle.

In order to assess the performance of PA-DLL, our software receiver i2xSNR was employed. The i2xSNR supports different code tracking architectures, including SDLL, VDLL, PA-DLL, and different positioning modes, including SPP, RTD, and RTK. The i2xSNR processes the IF data of GPS L1/BDS B1 with different DLL modes, respectively. To verify the impact of the NCO update rate on performance and processing load, both VDLL with an update rate of 50 Hz (VDLL-50 Hz) and 1 Hz (VDLL-1 Hz) were implemented and tested.

A vehicle test was carried out in downtown Shanghai, China. Figure 5 shows the trajectory of the test. The driving distance is about 15 km, including the viaduct, open sky, high buildings and trees, and glass curtain wall, as can be seen in Fig. 6.

Figure 7 shows the number of visible satellites and their $C/N_0$ during the test. The signal strength of most of the satellites decreased severely when the test vehicle drove under the viaduct (around second 115,000). Accordingly, the number of visible satellites was reduced to 4 and the GNSS positioning failed. In the situations of high buildings, trees, and glass curtain walls, the number of visible satellites was more than 12 during the test. However, the signal strength of most satellites changed frequently and the signal quality deteriorated significantly.

Integrated results of GNSS RTK and the navigation grade IMU with backward smoothing are served as the ground truth. GNSS base station was set up near the test area to collect data for GNSS RTK positioning. In addition, the pseudorange errors are evaluated using the pseudorange reference truth proposed by Feng et al. (2020).

## Test results and analysis

This section presents the field test results. We will analyze the performance of the SDLL, VDLL-1 Hz, VDLL-50 Hz, and PA-DLL in terms of code discriminator outputs,

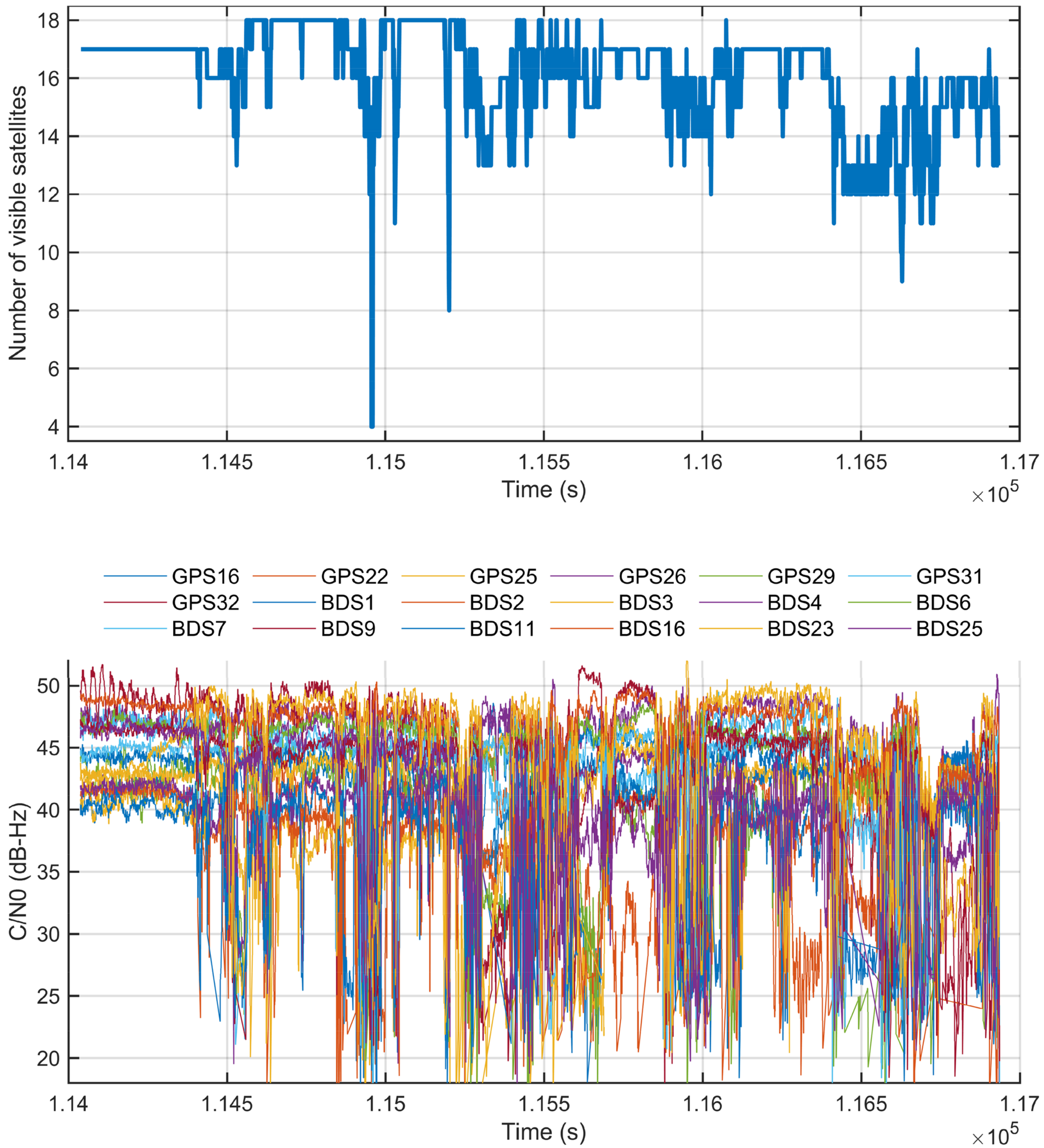



**Fig. 7** Number of visible satellites (top panel) and $C/N_0$ of all satellites (bottom panel). Complexity and challenge of the test are shown in the bottom panel

pseudorange errors and positioning errors. In addition, the processing load of four loops will be compared.

## Code discriminator results

Though the code discriminator output cannot accurately estimate the code phase error in complex environments, it can reflect the differences in NCO control information of the

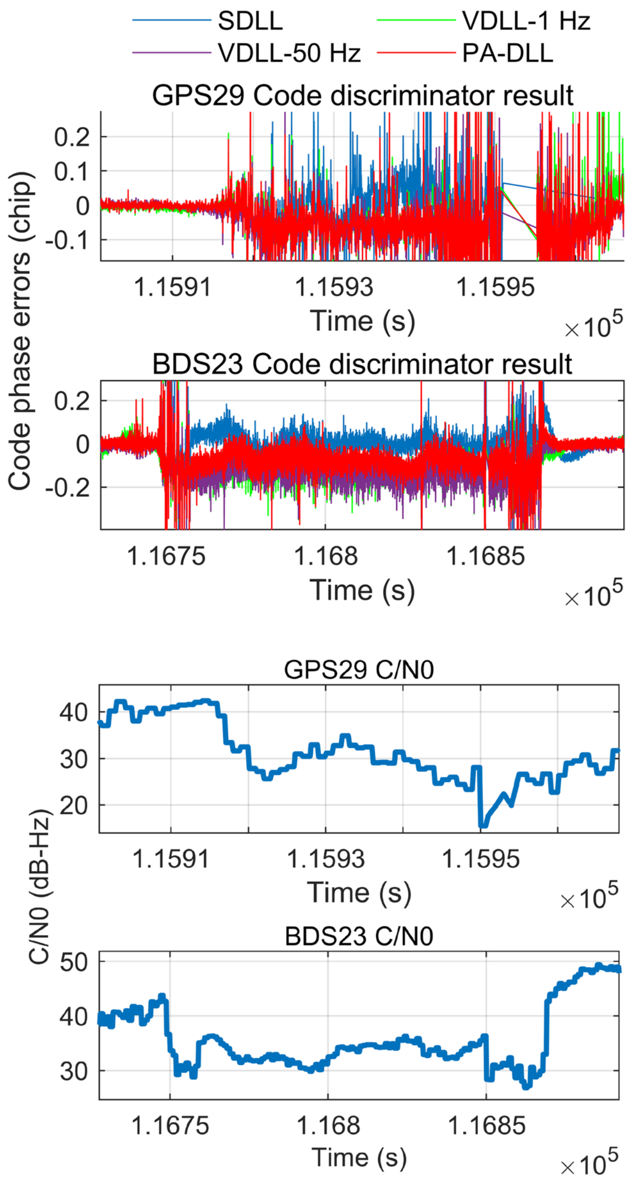



Fig. 8 Output of code discriminators (top panel) and $C/N_0$ in the corresponding time (bottom panel). The results of GPS29 in 115910–115960 s are corresponding to the situation of high buildings and trees in Fig. 6. Results of BDS23 correspond to the glass curtain wall in Fig. 6

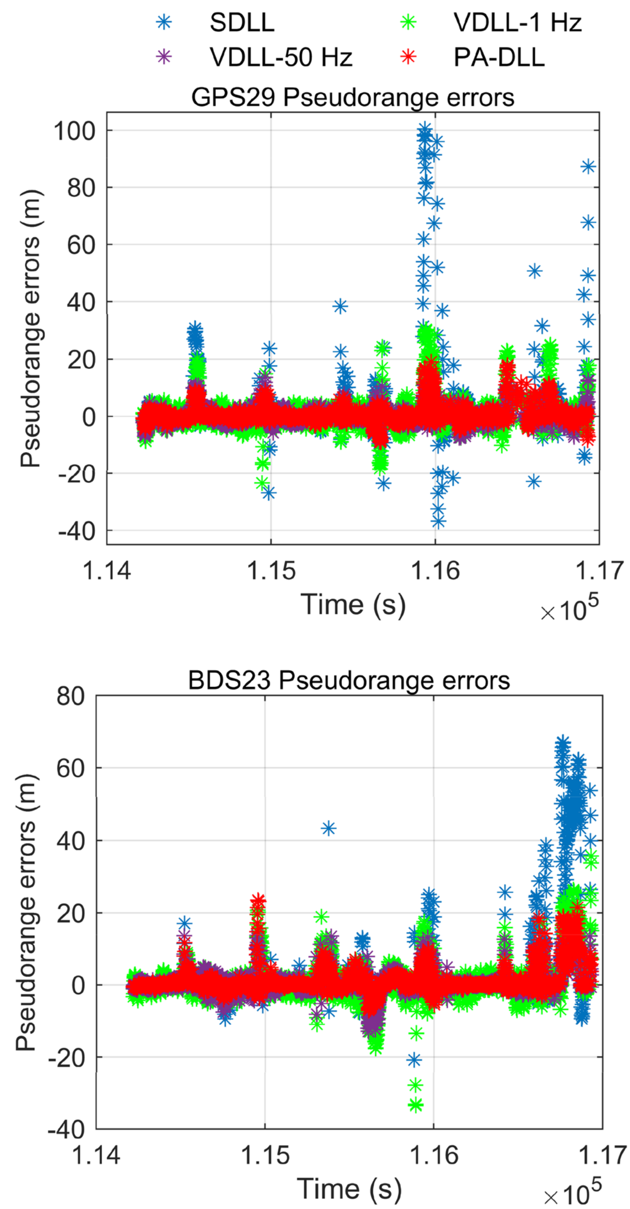



Fig. 9 Pseudorange error curves of the four code loops during the test

four code loops. Taking GPS29 and BDS23, for instance, Fig. 8 shows their $C/N_0$ values and discriminator results. It is seen from Fig. 8 that the discriminator results of PA-DLL are basically consistent with VDLL-50 Hz, while the discriminator output of the SDLL has an obvious difference with VDLL-50 Hz, and the results of VDLL-1 Hz show larger fluctuations. The signals of the two satellites are affected by fading and multipath. According to the study of Xu et al. (2020), the discriminator results of SDLL are affected by multipath, while VDLL reflects the code phase error of direct signal correctly. It is clear from Fig. 8 that PA-DLL has the same performance as VDLL. The discriminator results of PA-DLL and VDLL-50 Hz illustrate that their NCO control information is almost the same, which is consistent with the above NCO control information generation principle. The tracking performance of PA-DLL will be verified more intuitively by comparing pseudorange errors.

## Pseudorange errors

Figure 9 shows GPS29 and BDS23 pseudorange error curves of the four code loops. Compared with other loops, the pseudorange gross errors of SDLL are tens or even hundreds

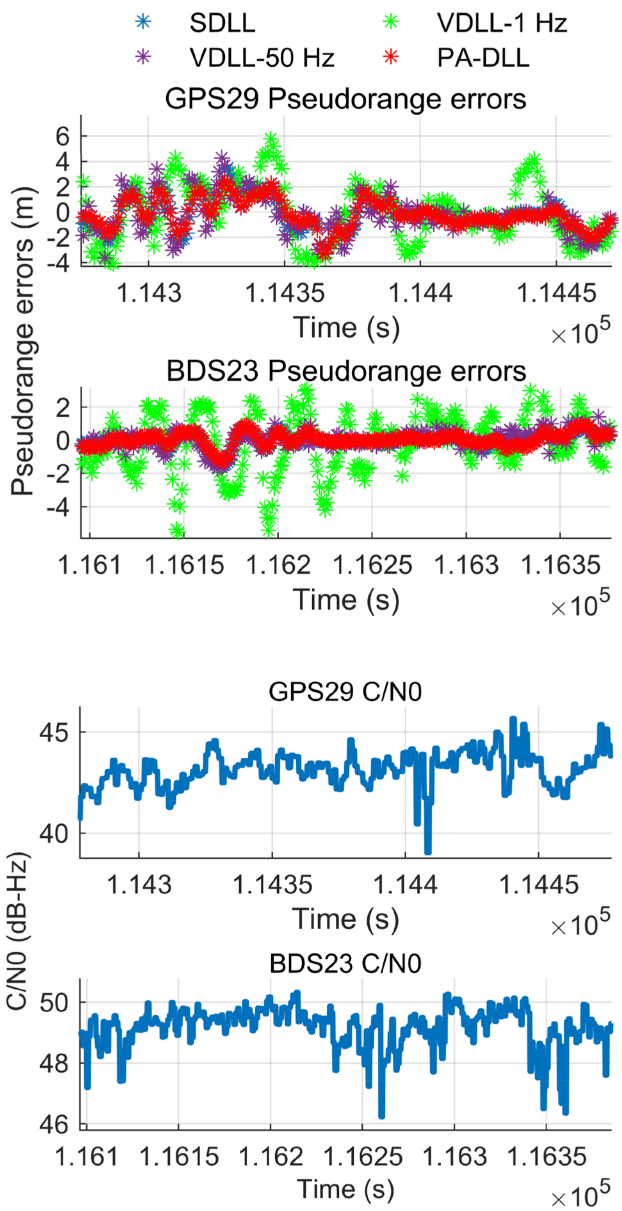



**Fig. 10** Pseudorange errors in the open sky (top panel) and $C/N_0$ (bottom panel) in the same period

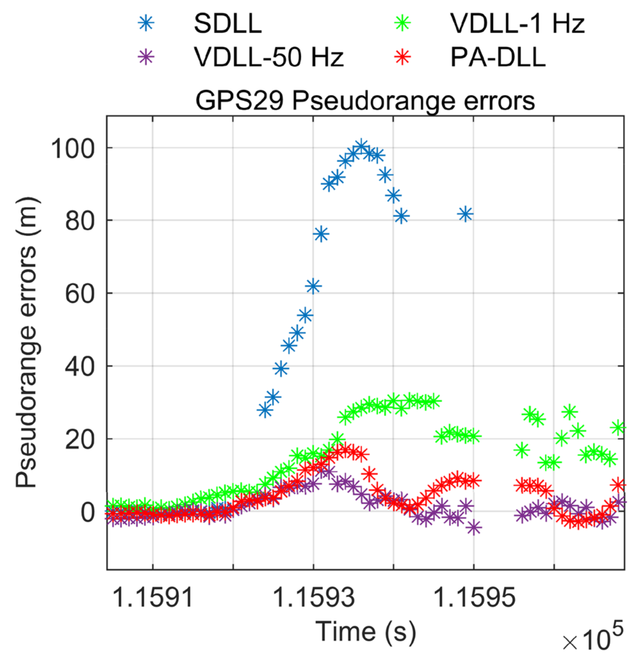



**Fig. 11** Pseudorange errors in signal weakened and multipath environments

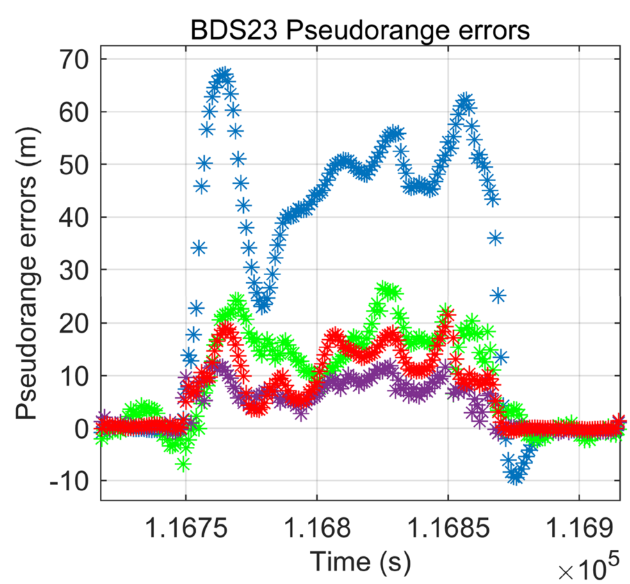



**Fig. 12** Pseudorange errors in multipath environments

of meters, which are much larger than that of VDLL and PA-DLL. The error curves of VDLL-50 Hz and PA-DLL are almost consistent, and the pseudorange errors are significantly suppressed. Though the gross errors of VDLL-1 Hz pseudorange are less than that of the SDLL, the accuracy is worse than that of VDLL-50 Hz and PA-DLL. To analyze the performance of the four tracking loops more specifically, the results in the open sky, signal blocked or weakened, multipath and GNSS denied environments are zoomed in and shown, respectively.

Under open sky, as shown in Fig. 6, the pseudorange errors are drawn in Fig. 10. The signal $C/N_0$ is almost larger than 40 dB-Hz, and the pseudorange noise of the SDLL, VDLL-50 Hz and PA-DLL is small and substantially equal. However, the pseudorange noise of VDLL-1 Hz is larger than the other three loops. A comparison of the pseudorange noise of PA-DLL and SDLL indicates that the aiding information estimated by positioning does not have negative

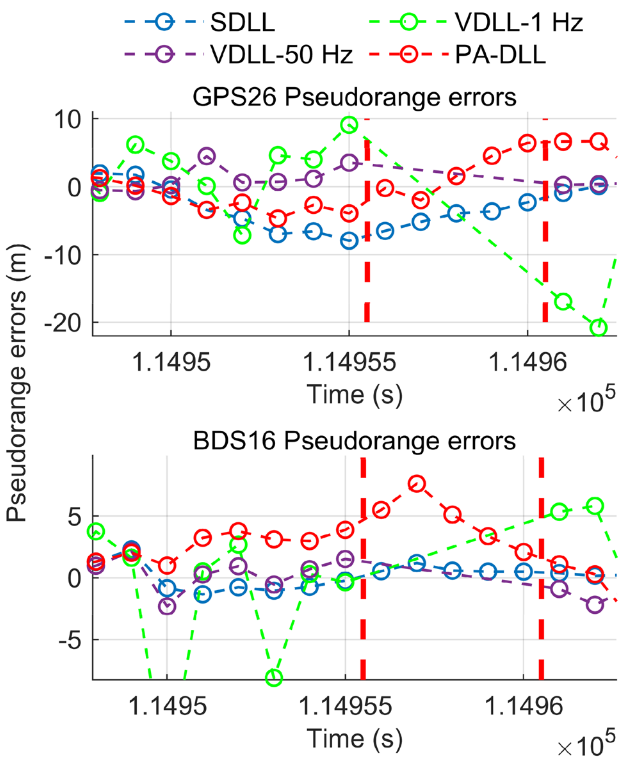



**Fig. 13** Pseudorange errors in GNSS-denied environments

effects in the open sky. The differences in pseudorange noise between VDLL-50 Hz and VDLL-1 Hz illustrate that the code NCO update rate can seriously affect the tracking accuracy of VDLL under dynamic vehicle conditions. The pseudorange noise consistency of PA-DLL and VDLL-50 Hz demonstrates that the performance of PA-DLL with 1 Hz positioning is equivalent to VDLL with 50 Hz positioning in the open sky.

When the satellite signals are blocked and weakened by high buildings and trees shown in Fig. 6, the pseudorange errors of four tracking loops are shown in Fig. 11. The pseudorange of SDLL is not continuous and has tens or even hundreds of meters error. The pseudorange error curves of PA-DLL and VDLL-50 Hz are almost coincident. Compared to SDLL, PA-DLL and VDLL-50 Hz can significantly improve the pseudorange accuracy and continuity. When the VDLL's NCO update rate is reduced to 1 Hz, the pseudorange errors show an obvious increase.

When the satellite signals are reflected by the glass curtain wall, as shown in Fig. 6, the pseudorange errors of four tracking loops are shown in Fig. 12. The pseudorange errors of SDLL increase to tens of meters, caused by multipath. Compared to SDLL, the pseudorange errors of PA-DLL, VDLL-1 Hz, and VDLL-50 Hz substantially reduce, and their pseudorange errors are comparable. Therefore, PA-DLL, VDLL-1 Hz and VDLL-50 Hz can suppress the multipath effects.

**Table 1** Statistical results of each satellite pseudorange errors (CEP95) and the proportion of different satellites during the test

| PRN | SDLL | | VDLL-1 Hz | | VDLL-50 Hz | | PA-DLL | |
|---|---|---|---|---|---|---|---|---|
| | Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] |
| GPS16 | 84 | 16.85 | 91 | 7.00 | 91 | 3.88 | 91 | **3.76** |
| GPS22 | 43 | 12.29 | 70 | 11.38 | 70 | 7.34 | 70 | **6.34** |
| GPS25 | 29 | **2.78** | 41 | 5.50 | 40 | 3.28 | 41 | 4.64 |
| GPS26 | 93 | 2.00 | 94 | 2.67 | 94 | **1.66** | 94 | 1.87 |
| GPS29 | 73 | 10.87 | 87 | 10.88 | 87 | **5.34** | 87 | 6.73 |
| GPS31 | 87 | 9.98 | 93 | 10.98 | 93 | 6.78 | 93 | **6.25** |
| GPS32 | 82 | 12.52 | 92 | 13.11 | 92 | 6.74 | 92 | **5.76** |
| BDS1 | 25 | **1.58** | 84 | 7.17 | 84 | 2.49 | 84 | 2.91 |
| BDS2 | 27 | 3.20 | 76 | 7.17 | 76 | **2.22** | 76 | 2.36 |
| BDS3 | 48 | 4.69 | 86 | 6.01 | 86 | **1.77** | 86 | 2.35 |
| BDS4 | 52 | 6.11 | 77 | 7.47 | 77 | **3.02** | 77 | 3.05 |
| BDS6 | 93 | 1.50 | 94 | 2.96 | 94 | 1.78 | 94 | **1.49** |
| BDS7 | 84 | 1.51 | 93 | 2.68 | 93 | **1.49** | 93 | 1.76 |
| BDS9 | 45 | **3.69** | 89 | 8.35 | 89 | 5.89 | 89 | 4.55 |
| BDS11 | 45 | 5.60 | 76 | 6.62 | 75 | 4.32 | 76 | **3.82** |
| BDS16 | 93 | **1.18** | 94 | 3.73 | 94 | 1.78 | 94 | 1.53 |
| BDS23 | 70 | 38.84 | 91 | 14.50 | 91 | **8.07** | 91 | 8.09 |
| BDS25 | 69 | 5.67 | 83 | 6.89 | 83 | **4.10** | 83 | 4.20 |

**Table 2** Pseudorange errors (CEP95) and valid epoch proportions of all satellites in total

| SDLL | | VDLL-1 Hz | | VDLL-50 Hz | | PA-DLL | |
|---|---|---|---|---|---|---|---|
| Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] | Valid epoch [%] | Error [m] |
| 63 | 5.84 | 84 | 7.29 | 84 | 3.97 | 84 | **3.90** |

When most of the satellite signals are blocked under the viaduct, as shown in Fig. 6, the GNSS positioning fails. Figure 13 shows the pseudorange error curves of visible satellites. As there is no position update, both VDLL-1 Hz and VDLL-50 Hz lose lock and cannot output pseudoranges. The SDLL is not affected by the positioning, so it can still output pseudoranges. When GNSS positioning fails, the PA-DLL degenerates into SDLL, which tracks the satellites with strong signal strength and outputs their pseudoranges. Therefore, the VDLL is constrained by the positioning, while the PA-DLL is not.

Overall, benefiting from multi-channel fusion and accurate aiding information, PA-DLL can always output continuous and accurate pseudoranges in open sky, and under signal-weakened, multipath and GNSS-denied situations. VDLL-50 Hz can output pseudoranges comparable with PA-DLL except for GNSS-denied environments. Limited by insufficient NCO update rate, the pseudorange accuracy of VDLL-1 Hz is worse than that of PA-DLL. Compared with PA-DLL, the accuracy and continuity of SDLL pseudorange seriously deteriorate in signal-weakened, frequently blocked, and multipath environments.

Tables 1 and 2 show the pseudorange valid epoch ratio and CEP95 error results of each visible satellite and the total statistical results of all visible satellites, respectively. The bold value denotes the minimum pseudorange error. While the valid epoch ratio of pseudorange from PA-DLL, VDLL-50 Hz and VDLL-1 Hz is about 83%, the ratio from SDLL is obviously less, only 63%. PA-DLL and VDLL have better tracking continuity in urban environments. The pseudorange CEP95 error statistics of all visible satellites show that, compared to the SDLL, the pseudorange errors of both PA-DLL and VDLL-50 Hz significantly reduce, and they are almost the same. However, the accuracy of VDLL's pseudorange drops a lot when its positioning update rate is down to 1 Hz. The statistical results indicate that the performance of PA-DLL is equal to or even better than that of VDLL in complex environments, which meets the expectation of the PA-DLL design.

## Position errors

In order to compare the pseudorange positioning performance of the four tracking loops fairly, a common positioning software (RTKLIB) is employed to process the pseudoranges from SDLL, VDLL-1 Hz, VDLL-50 Hz and PA-DLL.

Figure 14 shows positioning trajectories based on the four tracking loops in typical situations. In open sky, the four trajectories coincide well with the ground truth. In deep urban areas, the positioning results of SDLL are not continuous and deviate from the reference trajectory. However, the trajectories of VDLL-50 Hz and PA-DLL are still relatively smooth and nearly consistent with the reference.

Figure 15 shows the horizontal and vertical positioning errors and their corresponding cumulative distribution function (CDF). SDLL positioning errors are up to tens or even hundreds of meters, while most positioning errors of VDLL-50 Hz and PA-DLL are within 10 m. CDF curve in the horizontal direction demonstrates that SDLL and VDLL-1 Hz have 60.6% and 63.1% positioning errors within 5 m, respectively, while that of VDLL-50 Hz and PA-DLL are 79.0% and 80.4%, respectively. Compared to SDLL, the proportion of VDLL-50 Hz and PA-DLL nearly increases by 20%. SDLL positioning errors within 10 m account for 79.9%, and PA-DLL, VDLL-50 Hz and VDLL-1 Hz are 97.1%, 98.7% and 92.8%, respectively. Statistical results demonstrate that PA-DLL and VDLL-50 Hz can well suppress the errors of positioning and significantly improve the reliability of GNSS positioning.

## Processing load

Software receiver i2xSNR runs on a desktop computer with AMD Ryzen 7 3700X 4.4 GHz to test the processing load of the four tracking loops in this section. Time consumption of the four tracking loops was counted for processing 1 s of GNSS IF data, which is obtained by averaging the processing time of 400 s IF data, shown in Table 3. The bold value denotes the maximum time consumption within 1 s IF data. The processing load of SDLL is mainly from the discriminator and loop filter, and it only takes 0.01 ms to process 1 s IF data when the update rate is 50 Hz. Due to the heavy processing load of the positioning, the processing time of VDLL increases to 9.97 ms when the update rate is 1 Hz, and the time consumption sharply increases to 482 ms when the update rate is 50 Hz. Since the PA-DLL only needs one positioning per second, the time consumption is almost equivalent to that of VDLL-1 Hz. Positioning is a necessary unit in the GNSS receiver, and its update rate is usually 1 Hz. The total time consumption of loop and positioning in SDLL-based receiver is about 10 ms, which is equivalent to that of PA-DLL and VDLL-1 Hz. Therefore, there is little

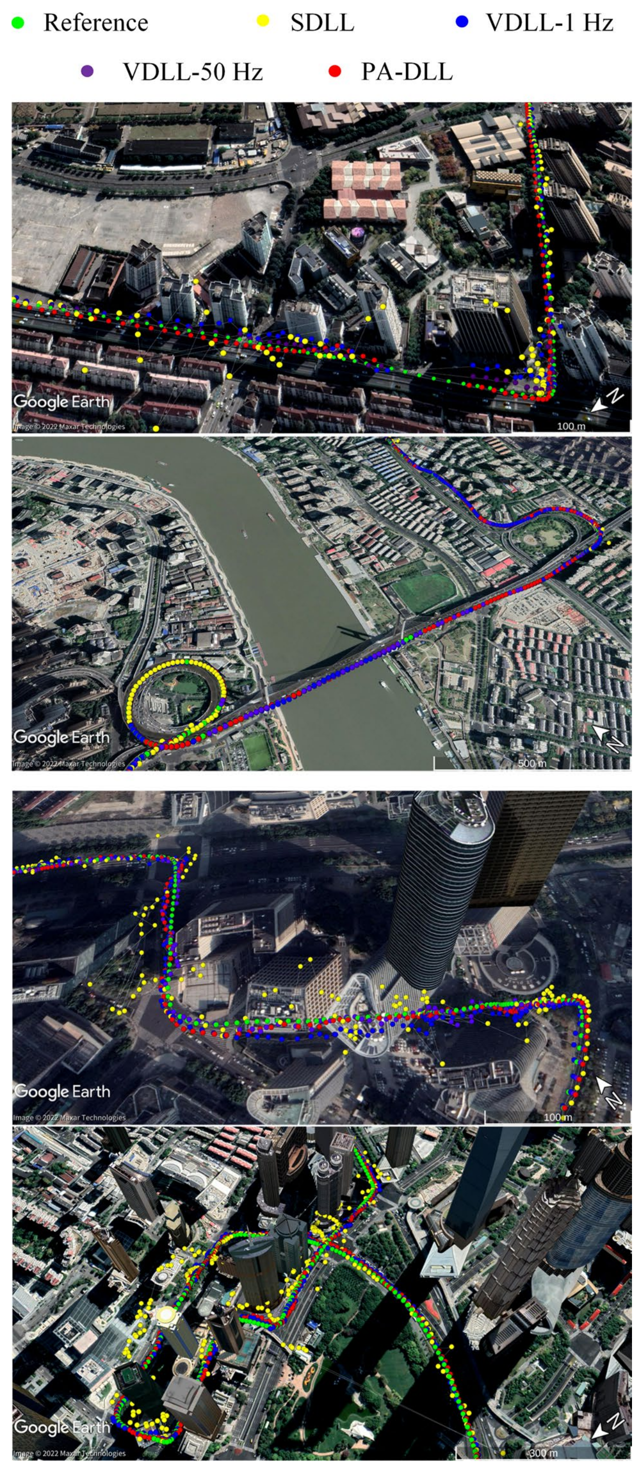



**Fig. 14** Different DLL's positioning trajectories compared to reference in four typical situations: under the viaduct (top left), open sky (top right), high buildings and trees (bottom left), glass curtain wall (bottom right) (from Google Earth)

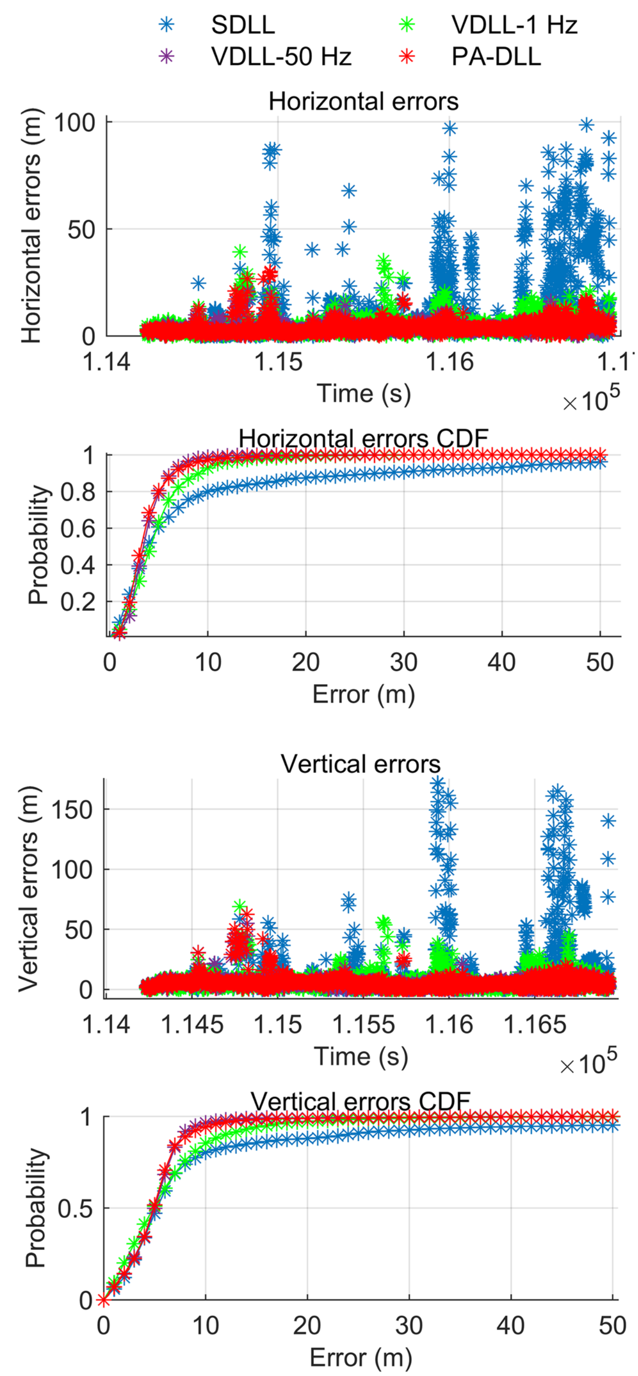



**Fig. 15** Horizontal and vertical positioning errors and their CDF

**Table 3** Different DLL's time consumption within 1 s IF data

| | SDLL | VDLL-50 Hz | VDLL-1 Hz | PA-DLL |
|---|---|---|---|---|
| Loop | 0.01 ms | **482.45** ms | 9.97 ms | 9.99 ms |
| Loop & positioning | 9.98 ms | **482.45** ms | 9.97 ms | 9.99 ms |

increase in processing load when adjusting from SDLL to PA-DLL in the receiver. To ensure the tracking and positioning performance, VDLL needs 10 or even dozens of times processing load with an update rate of no less than 10 Hz.

All the above results indicate that the performance of the proposed PA-DLL is comparable to that of VDLL with a high enough update rate in open sky, signal weakened and blocked, multipath environments and the PA-DLL can degenerate to SDLL in positioning denied areas. Meanwhile, PA-DLL can be implemented by only adding the code phase error estimation module to the SDLL-based receiver, with little increase in processing load.

## Conclusion

To ensure GNSS code tracking performance in challenging environments and be easy to implement in hardware receivers, we propose a PA-DLL. Compared to VDLL, PA-DLL is implemented with jointing estimated code phase error and the loop filter output to control code NCO and does not need to remove the existing SDLL in the receiver. PA-DLL architecture and key modules, including the EKF-based positioning and the aiding information estimation, are described. To verify PA-DLL's advantages in tracking performance and processing load, a vehicle test in urban environments was carried out, and the results were fully compared with SDLL and VDLL. Test results, including discriminator outputs, pseudorange error and positioning error, fully indicate that the performance of PA-DLL is equal to or even better than that of VDLL-50 Hz, and significantly better than that of SDLL in urban environments. Compared to SDLL, the pseudorange error of PA-DLL is reduced by 30%, and the positioning error of PA-DLL is reduced by 20%. In addition, when GNSS fails, PA-DLL degenerates into SDLL and still output pseudoranges. Furthermore, PA-DLL's update rate of positioning is reduced to 1 Hz compare to VDLL. There is little increase in processing load when adjusting SDLL to PA-DLL in the receiver. The proposed PA-DLL is helpful for the hardware receiver to improve tracking performance in challenging environments with minor adjustments.

**Acknowledgements** This research is funded by the National Key Research and Development Program of China (No. 2020YFB0505803), and the National Natural Science Foundation of China (No. 41974024).

**Data availability** This study has no associated data.

## References

Dardin SF, Calmettes V, Priot B, Tourneret J-Y (2013) Design of an adaptive vector-tracking loop for reliable positioning in harsh environment. In: Proceedings of ION GNSS 2013. Institute of Navigation, Nashville, Tennessee, USA, pp 3548–3559

Du S, Gao Y (2012) Inertial aided cycle slip detection and identification for integrated PPP GPS and INS. Sensors 12:14344–14362

Feng X, Zhang T, Lin T, Tang H, Niu X (2020) Implementation and performance of a deeply-coupled GNSS receiver with low-cost MEMS inertial sensors for vehicle urban navigation. Sensors 20:3397

Gomez-Ojeda R, Moreno F-A, Zuniga-Noël D, Scaramuzza D, Gonzalez-Jimenez J (2019) PL-SLAM: a stereo SLAM system through the combination of points and line segments. IEEE Trans Robot 35:734–746

Groves PD (2015) Principles of GNSS, inertial, and multisensor integrated navigation systems, [Book review]. IEEE Aerosp Electron Syst Mag 30:26–27

Groves PD, Mather CJ (2010) Receiver interface requirements for deep INS/GNSS integration and vector tracking. J Navig 63:471–489. https://doi.org/10.1017/S0373463310000093

Gustafson D, Dowdle J, Flueckiger K (2000) A deeply integrated adaptive GPS-based navigator with extended range code tracking. In: IEEE 2000. Position Location and Navigation Symposium (Cat. No. 00CH37062). IEEE, pp 118–124

Hsu L-T, Jan S-S, Groves PD, Kubo N (2015) Multipath mitigation and NLOS detection using vector tracking in urban environments. GPS Solut 19:249–262. https://doi.org/10.1007/s10291-014-0384-6

Hsu L-T, Gu Y, Kamijo S (2016) 3D building model-based pedestrian positioning method using GPS/GLONASS/QZSS and its reliability calculation. GPS Solut 20:413–428

Jiang C, Xu B, Hsu L-T (2021) Probabilistic approach to detect and correct GNSS NLOS signals using an augmented state vector in the extended Kalman filter. GPS Solut 25:1–14

Jiao C, Wang X, Wang D, Li Q, Zhang J, Cai Y (2021) An adaptive vector tracking scheme for high-orbit degraded GNSS signal. J Navig 74:105–124

Ju B, Gu D, Chang X, Herring TA, Duan X, Wang Z (2017) Enhanced cycle slip detection method for dual-frequency BeiDou GEO carrier phase observations. Gps Solut 21:1227–1238

Kaplan ED, Hegarty C (2017) Understanding GPS/GNSS: principles and applications. Artech House, Massachusetts

Lashley M, Bevly D, Petovello M (2009) What are vector tracking loops, and what are their benefits and drawbacks? GNSS 4:16–21

Lashley M, Bevly DM, Hung JY (2010) A valid comparison of vector and scalar tracking loops. IEEE/ION position, location and navigation Symposium. IEEE, Indian Wells, CA, pp 464–474

Li T, Zhang H, Gao Z, Chen Q, Niu X (2018) High-accuracy positioning in urban environments using single-frequency multi-GNSS RTK/MEMS-IMU integration. Remote Sens 10:205

Liu Y, Li S, Fu Q, Liu Z, Zhou Q (2019) Analysis of Kalman filter innovation-based GNSS spoofing detection method for INS/GNSS integrated navigation system. IEEE Sens J 19:5167–5178. https://doi.org/10.1109/JSEN.2019.2902178

Pany T, Eissfeller B (2006) Use of a vector delay lock loop receiver for GNSS signal power analysis in bad signal conditions. Proc IEEE/ION Plans 2006:893–903

Parkinson BW, Spilker JJ (1996) Progress in astronautics and aeronautics: global positioning system: theory and applications. Aiaa, Reston

Parkinson B (2014) Assured PNT for our future: PTA. Actions necessary to reduce vulnerability and ensure availability. GPS World 14 (9):1–10

Petovello MG, O'Driscoll C, Lachapelle G (2008) Carrier phase tracking of weak signals using different receiver architectures. In: Proceedings of ION NTM 2008, Institute of Navigation, San Diego, CA, USA, January 28 - 30, pp 781–791

Ren T, Petovello MG (2017) A stand-alone approach for high-sensitivity GNSS receivers in signal-challenged environment. IEEE Trans Aerosp Electron Syst 53:2438–2448. https://doi.org/10.1109/TAES.2017.2699539

Teunissen PJ, Kleusberg A (2007) GPS observation equations and positioning concepts. In: GPS for Geodesy, Springer, pp 175–217

Wang Y, Li X, Khoshelham K, Li P (2019) Robust iterated extended Kalman filter algorithm for foot-mounted inertial measurement units/ultrawideband fusion positioning. J Appl Remote Sens 13:1. https://doi.org/10.1117/1.JRS.13.024510
Xu Q, Li X, Chan C-Y (2018) Enhancing localization accuracy of MEMS-INS/GPS/in-vehicle sensors integration during GPS outages. IEEE Trans Instrum Meas 67:1966–1978
Xu B, Jia Q, Hsu L-T (2020) Vector tracking loop-based GNSS NLOS detection and correction: algorithm design and performance analysis. IEEE Trans Instrum Meas 69:4604–4619. https://doi.org/10.1109/TIM.2019.2950578
Zhang Z, Li B, Gao Y, Shen Y (2019) Real-time carrier phase multipath detection based on dual-frequency C/N0 data. GPS Solut 23:1–13
Zhang G, Xu P, Xu H, Hsu L-T (2021a) Prediction on the urban GNSS measurement uncertainty based on deep learning networks with long short-term memory. IEEE Sens J 21:20563–20577
Zhang Z, Niu X, Tang H, Chen Q, Zhang T (2021b) GNSS/INS/ODO/wheel angle integrated navigation algorithm for an all-wheel steering robot. Meas Sci Technol 32:115122

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

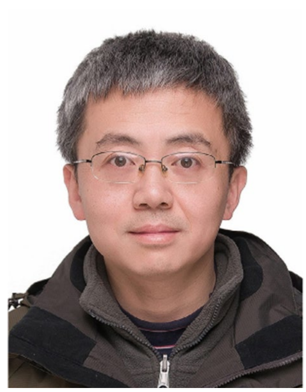


**Tisheng Zhang** is an associate professor in GNSS Research Center at Wuhan University, China. He holds a B.SC. and a Ph.D. in Communication and Information Systems from Wuhan University, Wuhan, China, in 2008 and 2013, respectively. From 2018 to 2019, he was a PostDoctor of the Hong Kong Polytechnic University. His research interests focus on the fields of GNSS receiver, deep multisensor integration and robot navigation. He has published more than 40 academic papers and own 16 patents.

**Jinwei Shi** received a B.Eng. degree in electronic information engineering from Wuhan University, Wuhan, China, in 2020, where he is currently pursuing a master's degree in navigation, guide, and control with the GNSS Research Center. His primary research interest focuses on deeply coupled GNSS receiver.

**Tao Lin** is currently a systems architect at UniStrong. He is responsible for GNSS signal processing, positioning algorithms, system design and technology development. He received his B.Sc. and Ph.D. in Geomatics Engineering from the University of Calgary and worked as a postdoctoral fellow and senior research associate in the PLAN group from 2013 to 2017.

**Xin Feng** received the B.E and M.E degrees from Wuhan University, Wuhan, China, in 2018 and 2021, respectively. He is pursuing a Ph.D. in geodesy and survey engineering with the School of Geodesy and Geomatics at Wuhan University. His primary research interests include GNSS receiver and multisensor ultra-tightly integrations.



**Xiaoji Niu** is a Professor at the GNSS Research Center, Wuhan University, China. He has published more than 90 academic papers and owns 28 patents. He leads a multisensor navigation group focusing on GNSS/INS integration, low-cost navigation sensor fusion, and its new applications.