Chinese Journal of Aeronautics, (2024), 37(4): 405–420

CSAA

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

FULL LENGTH ARTICLE

A robust vector tracking loop structure based on potential bias analysis

Qiongqiong JIA [a], Yiran LUO [b,c], Bing XU [b], Li-Ta HSU [b], Renbiao WU [a,*]

[a] Tianjin Key Lab for Advanced Signal Processing, Civil Aviation University of China, Tianjin 300300, China
[b] Interdisciplinary Division of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, 999077, Hong Kong, China
[c] Department of Geomatics Engineering, University of Calgary, Calgary T2N1N4, Canada

Received 6 April 2023; revised 11 June 2023; accepted 8 September 2023
Available online 22 November 2023

KEYWORDS

Global Navigation Satellite System (GNSS);
Vector Tracking Loop (VTL);
Multipath;
Non-Line-of-Sight (NLOS);
Tracking bias propagation;
Least Absolute Shrinkage and Selection Operator (LASSO)

Abstract This paper proposes a robust vector tracking loop structure based on potential bias analysis. The influence of four kinds of biases on the existing two implementations of Vector Tracking Loops (VTLs) is illustrated by theoretical analysis and numerical simulations, and the following findings are obtained. Firstly, the initial user state bias leads to steady navigation solution bias in the relative VTL, while new measurements can eliminate it in the absolute VTL. Secondly, the initial code phase bias is transferred to the following navigation solutions in the relative VTL, while new measurements can eliminate it in the absolute VTL. Thirdly, the user state bias induced by erroneous navigation solution of VTLs can be eliminated by both of the two VTLs. Fourthly, the multipath/NLOS likely affects the two VTLs, and the induced tracking bias in the duration of the multipath/NLOS would decrease the performance of VTLs. Based on the above analysis, a robust VTL structure is proposed, where the absolute VTL is selected for its robustness to the two kinds of initialization biases; meanwhile, the instant bias detection and correction method is used to improve the performance of VTLs in the duration of the multipath/NLOS. Numerical simulations and experimental results verify the effectiveness of the proposed robust VTL structure.

1. Introduction

Global Navigation Satellite System (GNSS) is widely applied in various aspects of our modern lives. Most GNSS receivers utilize the Scalar Tracking Loop (STL) to track incoming signals and generate pseudorange and pseudorange rate measurements individually, i.e., one tracking channel for one satellite signal.[1–2] Due to its simplicity in implementation and effectiveness in benign scenarios, the STL is still the most popular GNSS receiver structure. However, the performance of STLs is easily degraded in harsh environments.[3–5]

[*] Corresponding author.
E-mail address: rbwu@cauc.edu.cn (R. WU).

Peer review under responsibility of Editorial Committee of CJA.

ELSEVIER Production and hosting by Elsevier

https://doi.org/10.1016/j.cja.2023.11.013
1000-9361 © 2023 Production and hosting by Elsevier Ltd. on behalf of Chinese Society of Aeronautics and Astronautics.

By utilizing the fact that all the in-view satellites are interconnected through one common receiver, the Vector Tracking Loop (VTL) provides a deep integration of signal tracking and navigation solving.[6–8] The mostly cited advantages of VTLs are the increased immunity to interference, high sensitivity for weak signal tracking, continuous tracking for signals that are temporary blocked, and immediate re-lock of signals.

Despite the above-mentioned advantages, VTLs suffer from fundamental limitations. First, VTLs are usually initialized by the STL or other exterior sources, which might induce initialization biases such as the initial user state bias and initial code phase bias. Second, the biases during the operational period of VTLs, e.g., the erroneous navigation solution induced user state bias and the multipath/NLOS reception induced tracking bias might also degrade the performance of VTLs. Besides, the internal coupling of VTLs also causes bias propagation among tracking channels.[9–11]

1.1. Related work

The initial user state bias induced navigation solution deviation was studied in Ref. 12. According to this study, the primary reason for the deviation is that in the existing VTL implementation, the measurements of the VTL are coupled with the user state increment in consecutive epochs. Hence, the navigation solution calculated from the measurements reflects the user state increment instead of the absolute user state.[13–14] Therefore, the user state bias cannot be eliminated in the following navigation solutions. We termed this conventional VTL implementation as the relative VTL. The absolute VTL implementation was proposed in Ref. 11 to deal with the above problems of the initial user state bias and the initial code phase bias. However, the difference between the initial biases and the biases during the operational period of VTLs has not been illustrated.

To overcome the multipath/NLOS-induced tracking bias, the multiple Fault Detection and Exclusion (FDE) algorithms are introduced to detect and exclude the affected channel from the navigation filter. The carrier-to-noise density ratio (C/N_0) is an important indicator of signal quality, which is used to detect blocked channels.[15] The performance of several commonly used methods for calculating C/N_0 in VTLs is compared, and an improved method, the Receiver Autonomous Integrity Monitoring (RAIM), is proposed to smooth the noise adaptively in Ref. 15. RAIM utilizes redundant measurements for fault detection, by which potential threats can be identified.[16] Therefore, the RAIM is a very useful technique to prevent the failure of one channel from spreading into other tracking channels. Extensive studies of using RAIM algorithms for VTLs can be found in previous studies.[16–18] The noise bandwidth of VTLs, which was firstly put forward in Ref. 19 was then extended as a more compact and general form in Ref. 20. The noise bandwidth is modeled as a function of the C/N_0 from individual channels, the relative C/N_0 across different tracking channels, and the user-satellite geometry.[20] In Ref. 21, the noise bandwidth is utilized to identify the multipath contaminated channels, and enhanced performance is obtained by discarding the detected channels. Although the above FDE methods are effective in preventing bias propagation in VTLs, they cannot be directly used in the urban environment where the number of visible satellites is limited, as exclusion of the affected satellite signals may result in insufficient number of satellites or a bad geometry for positioning.[22]

To make full use of low-quality signals in urban areas and to reduce the bias propagation of VTLs, a diagonal weighting technique is proposed,[23] where the off-diagonal components of the weighting matrix are set to zero. However, as the off-diagonal components denote the cross correlations among tracking channels, zero components mean that the internal aiding of VTL is eliminated, hence the superiority of VTLs is also vanished. To alleviate the effect of low-quality signals, a covariance rescaling method is proposed,[24] where a linear local filter and an adaptive navigation filter is designed to calculate the rescaling factor. However, if the tracking performance of one channel is degraded for signal quality decline, e.g., multipath or NLOS-reception, etc., both the local filter and the global filter diverges. Besides, the local filters used for all the tracking channels also increase the complexity of the receiver structure. Based on the existing multipath detection methods for the VTL,[25–26] an adaptive STL-VTL structure and a conjoint STL-VTL scheme are proposed to reduce the multipath effect. The adaptive STL-VTL structure can smoothly switch between each other, while the conjoint STL-VTL contains both STL and VTL modes and selects the healthy measurements based on the multipath detection result. The adaptive and conjoint STL-VTL schemes take the advantages of both STLs and VTLs regarding positioning reliability and tracking robustness.[24]

Another category of methods for dealing with the VTL tracking bias is to find more reliable measurements to replace the ones in the degraded channels. One kind of these method is the incorporation of neural network into VTLs.[27–29] The neural network is trained when the signals are in good conditions, and is then applied in the duration of signal blockage. A strobe correlator is also proposed to decrease the bias propagation of VTLs;[30] however, this method is not able to deal with the NLOS-induced tracking bias. To deal with the NLOS-induced tracking bias, the code discriminator output is applied to estimate the bias, and then the estimates are subtracted from the measurements.[10] Whereas, the linear relation between the discriminator output and the NLOS-induced bias only holds within the linear scope of the discriminate curve. Besides, the tracking bias estimation is based on the sample average of the discriminator output, which means that the tracking bias has already propagated before a reliable estimation can be obtained. Therefore, the bias correction performance is decreased. In Refs. 31–32, an augmented state vector for the navigation filter is created by adding the NLOS-induced bias to the state vector, then the NLOS-induced pseudorange measurement bias is estimated as a state variable. However, adding unknowns to the sate vector means increasing the minimum requirement of the measurements, which might be difficult in the constrained environment such as in urban canyon.

1.2. Contributions of this work

Four potential bias sources (including the initial user state bias, the initial code phase bias, the erroneous navigation solution bias induced user state bias, and the multipath/NLOS induced tracking bias) and two existing VTL implementations are discussed firstly, and the following conclusions are drawn: (A) The initial user state bias results in steady bias in the following navigation solutions in the relative VTL, while it can be eliminated by the new measurements in the absolute VTL; (B) The initial code phase bias would be transferred to a bias in the following navigation solutions in the relative VTL, but can be eliminated in the absolute VTL; (C) The erroneous user state bias during VTLs' operational period can be eliminated by both of the two VTL implementations; (D) The multipath/NLOS likely affects both of the VTLs, specifically, the multipath/NLOS introduces tracking bias, and furthers the bias propagation, but the induced bias disappears as the signal recovers; (E) Bias propagation in VTLs would decrease the bias correction performance in the duration of the multipath/NLOS. Hence, to ensure the performance of VTLs, it is important to correct the tracking bias before its propagation.

Based on the above conclusions, a VTL structure which is robust to the above-mentioned biases is proposed, where the absolute VTL is selected for its robustness to the two initialization biases, and the instant bias detection and correction method is used to eliminate the multipath/NLOS introduced tracking bias in its duration. The measurement residuals-based statistic is established to detect the multipath/NLOS induced tracking bias in the proposed way. Once a tracking bias is detected, a Least Absolute Shrinkage and Selection Operator (LASSO) based bias estimation method is turned on. Then, the original measurement is corrected by the estimated bias. Numerical simulations and experimental results verify the effectiveness of the proposed robust VTL structure.

The paper is organized as follows. The VTL basics and two existing implementations of VTL are given in Section 2. Robustness of the two VTL implementations to four biases are discussed in Section 3. The proposed robust VTL structure is given in Section 4. The experimental results are given in Section 5. Finally, the conclusions are given in Section 6.

Throughout the paper, the following notations are used unless specifically stated:

$\boldsymbol{a}$: bold-face lower-case letters denote column vectors.

$\boldsymbol{A}$: bold-face capital letters denote matrices.

a^m: the superscript m stands for the satellite index.

a_k: the subscript k denotes the epoch index.

a^-, a^+: the superscript "$-$" denotes the priori prediction from the EKF, while "$+$" denotes the corresponding posterior estimation.

$\hat{a}$: the estimation of a.

δa: the residual, i.e., $\delta a = a - a^-$.

δa^+: the residual, i.e., $\delta a = a - a^+$.

Δa: the increment of a in successive epochs, i.e., $\Delta a_k = a_k - a_{k-1}$.

$(\cdot)^{\mathrm{T}}$: the transpose of a vector or a matrix.

a': the biased quantity originated from the initial user state bias.

a'': the biased quantity originated from the initial code phase bias.

a''': the biased quantity originated from user state bias in the VTLs' operational period.

a'''': the biased quantity originated from the tracking bias in the VTLs' operational period.

$\breve{a}$: after bias correction.

$\| \cdot \|_{\mathrm{p}}$: the l_{p}-norm.

2. VTL basics

The focus of this section is to present the basic principle of GNSS and two VTL implementations.

2.1. GNSS basics

Define $\boldsymbol{x} = [\boldsymbol{p}, ct_b]^{\mathrm{T}}$ as the user state, where $\boldsymbol{p} = [p_x, p_y, p_z]^{\mathrm{T}}$ stands for the 3-D user position in the Earth-Centered Earth-Fixed (ECEF) coordinate system. The variable t_b denotes the user clock bias, with c standing for the speed of light. The navigation solution is obtained by measuring the pseudorange between the user receiver and satellites. The pseudorange measurements of the m-th satellite can be formulated by[1]

$$\rho^m = \| \boldsymbol{p} - \boldsymbol{p}^m \|_2 + ct_{\mathrm{b}} - ct_{\mathrm{b}}^m + I^m + T^m + \varepsilon^m \tag{1}$$

where $\boldsymbol{p}^m$ stands for the 3-D position, t_{b}^m denotes the clock bias, I^m and T^m are the ionosphere and troposphere delay of the m-th satellite, respectively, which can be obtained from the ephemeris; the variable ε^m denotes the measurement noise of the m-th tracking channel. Arranging the pseudorange measurements from all the tracking channels as a compact vector $\boldsymbol{\rho} = [\rho^1, \rho^2, ..., \rho^M]^{\mathrm{T}}$, the relationship between the measurements and the user state is formulated as below

$$\boldsymbol{\rho} = h(\boldsymbol{x}) + \boldsymbol{\varepsilon} \tag{2}$$

where $h(\cdot)$ stands for the observation function,[1] and $\boldsymbol{\varepsilon} = [\varepsilon^1, \varepsilon^2, ..., \varepsilon^M]^{\mathrm{T}}$ is the measurement noise vector.

The code-phases are measured depending on the timing of the 'time-of-reception' hardware counter. The time of propagation is obtained by comparing the signal transmission time with the local receiver time. With the time of propagation, the pseudorange measurement is calculated by[2,9]

$$\boldsymbol{\rho}_k = c(\boldsymbol{t}_{\mathrm{or},k} - \hat{\boldsymbol{\tau}}_k) = g(\hat{\boldsymbol{\tau}}_k) \tag{3}$$

where the subscript k denotes the epoch index. Since $\boldsymbol{t}_{\mathrm{or},k}$ can be read from the receiver directly, the measurement is expressed as a function of the signal transmission time $\hat{\boldsymbol{\tau}}_k$, as the second equation in Eq. (3). The transmission time $\hat{\boldsymbol{\tau}}_k$ is obtained by the code phases from tracking loops. Using true parameters to replace the estimated ones in Eq. (3), we have the relation below

$$\boldsymbol{\rho}_k = g(\boldsymbol{\tau}_k) + \boldsymbol{\varepsilon}_k = h(\boldsymbol{x}_k) + \boldsymbol{\varepsilon}_k \tag{4}$$

The a priori user state prediction can be represented as[13–14]

$$\boldsymbol{x}_k^- = \boldsymbol{x}_{k-1}^+ + \Delta \boldsymbol{x}_k^- \tag{5}$$

where the user state increment prediction with the constant velocity dynamic model is

$$\Delta \boldsymbol{x}_k^- = T \boldsymbol{v}_{k-1}^+ \tag{6}$$

where T is the epoch length; $\boldsymbol{v}_{k-1}^+$ denotes a posterior estimation of $\boldsymbol{v}_{k-1}$, which is composed of the 3-D user velocity and the clock drift. Then, the first order Taylor expansion of Eq. (2) nearby $\boldsymbol{x}_k^-$ can be represented as[9,33]

$$\boldsymbol{\rho}_k \approx h\left(\boldsymbol{x}_k^-\right) + \boldsymbol{H}_k\left(\boldsymbol{x}_k - \boldsymbol{x}_k^-\right) + \boldsymbol{\varepsilon}_k \tag{7}$$

where $\boldsymbol{H}_k$ is the Jacobian of $h(\cdot)$, which is also termed as the measurement matrix. Define the measurement residuals as

$$\delta\boldsymbol{\rho}_k = \boldsymbol{\rho}_k - h\left(\boldsymbol{x}_k^-\right) = \boldsymbol{H}_k\delta\boldsymbol{x}_k + \boldsymbol{\varepsilon}_k \tag{8}$$

where $\delta\boldsymbol{x}_k$ is the user state residual. By inserting Eq. (4) into Eq. (8), we have

$$\delta\boldsymbol{\rho}_k = c\delta\boldsymbol{\tau}_k + \boldsymbol{\varepsilon}_k \tag{9}$$

The code discriminator output the code phase residuals $\delta\boldsymbol{\tau}_k$. Considering the relations between the output of the discriminators and the measurements of VTLs in Eq. (9), the discriminator output is also termed as the measurement in this work for convenience.

With the pseudorange residuals $\delta\boldsymbol{\rho}_k$, the user state residual estimation $\delta\hat{\boldsymbol{x}}_k$ can be obtained by the EKF. Finally, the posterior user state, which is also termed as the navigation solution, can be updated by

$$\boldsymbol{x}_k^+ = \boldsymbol{x}_k^- + \delta\hat{\boldsymbol{x}}_k \tag{10}$$

Eq. (10) can be interpreted as the convergence of the a priori predicted user state $\boldsymbol{x}_k^-$ to the true user state $\boldsymbol{x}_k$ by adding the user state residual estimation $\delta\hat{\boldsymbol{x}}_k$.

2.2. Relative VTL implementation

In the traditional VTL implementation, the code phase of the local replica is calculated in a relative manner, where the code phase increment is calculated with the user state increment by[14]

$$\Delta\boldsymbol{\tau}_k^- = \boldsymbol{H}_k\Delta\boldsymbol{x}_k^-/c \tag{11}$$

Then, the code phase of the local replica is updated by

$$\boldsymbol{\tau}_k^- = \boldsymbol{\tau}_{k-1}^+ + \Delta\boldsymbol{\tau}_k^- = h\left(\boldsymbol{x}_{k-1}^+\right)/c + \boldsymbol{H}_k\Delta\boldsymbol{x}_k^-/c = h\left(\boldsymbol{x}_k^-\right)/c \tag{12}$$

The first equation reveals the code phase adjustment manner in the relative VTL. Specifically, the code phase is adjusted by adding the code phase increment prediction to the posterior code phase of the previous epoch. The last two equations in Eq. (12) are based on the relations in Eq. (4), which are given to illustrate the corresponding relations among the user states.

To get the measurement in the relative VTL, we replace the predicted quantities in Eq. (12) with the corresponding true values carried by the received signal as

$$\boldsymbol{\tau}_k = \boldsymbol{\tau}_{k-1} + \Delta\boldsymbol{\tau}_k = h(\boldsymbol{x}_{k-1})/c + \boldsymbol{H}_k\Delta\boldsymbol{x}_k/c = h(\boldsymbol{x}_k)/c \tag{13}$$

where $\Delta\boldsymbol{x}_k = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$. With Eqs. (12) and (13), the code phase residual is represented as

$$\begin{aligned}\delta\boldsymbol{\tau}_k &= \boldsymbol{\tau}_k - \boldsymbol{\tau}_k^- = \delta\boldsymbol{\tau}_{k-1}^+ + \delta(\Delta\boldsymbol{\tau}_k) = \boldsymbol{H}_k(\delta\boldsymbol{x}_{k-1}^+ + \delta(\Delta\boldsymbol{x}_k))/c\\ &= \boldsymbol{H}_k\delta\boldsymbol{x}_k/c\end{aligned} \tag{14}$$

It is noted from Eq. (14) that the code phase residual $\delta\boldsymbol{\tau}_k$ is coupled with the summation of the posterior user state increment of the previous epoch $\delta\boldsymbol{x}_{k-1}^+$ and the user state increment residual $\delta(\Delta\boldsymbol{x}_k)$. The summation is also equal to the user state residual $\delta\boldsymbol{x}_k$, shown as the last equation in Eq. (14). With the code phase residuals as the measurements in Eq. (14), the EKF outputs the user state residual estimation $\delta\hat{\boldsymbol{x}}_k$, which is not only used to obtain the navigation solution by Eq. (10), but also used to calculate the posterior code phase adjustment as[14]

$$\delta\hat{\boldsymbol{\tau}}_k = \boldsymbol{H}_k\delta\hat{\boldsymbol{x}}_k/c \tag{15}$$

Then, the prior predicted code phase in Eq. (12) is corrected as

$$\boldsymbol{\tau}_k^+ = \boldsymbol{\tau}_k^- + \delta\hat{\boldsymbol{\tau}}_k \tag{16}$$

2.3. Absolute VTL implementation

Since the code phase of a satellite signal is directly related with its transmission time; therefore, once the transmission time is obtained, the code phase can be determined. The relation between code phase prediction and user state prediction is formulated as[11]

$$\boldsymbol{\tau}_k^- = g^{-1}\left(h\left(\boldsymbol{x}_k^-\right)\right) \tag{17}$$

Similarly, the relation between the true code phase of the received signal and the corresponding true user state is

$$\boldsymbol{\tau}_k = g^{-1}(h(\boldsymbol{x}_k)) \tag{18}$$

The true quantities in Eq. (18) cannot be obtained in reality. Actually, only the predicted and the estimated ones can be obtained, while the true quantities are used to illustrate the relations among the quantities. The code phase difference between the true code phase carried by the received signal and the predicted one can be obtained by the code discriminator. With Eqs. (17) and (18), the code phase residual in the absolute VTL is

$$\delta\boldsymbol{\tau}_k = g^{-1}(h(\boldsymbol{x}_k)) - g^{-1}\left(h\left(\boldsymbol{x}_k^-\right)\right) = \boldsymbol{H}_k\delta\boldsymbol{x}_k/c \tag{19}$$

With the measurement in Eq. (19), the user state residual estimation $\delta\hat{\boldsymbol{x}}_k$ and further the navigation solution $\boldsymbol{x}_k^+$ can be updated. It is noted from Eq. (17) that the code phase is calculated by the absolute user state prediction of the current epoch $\boldsymbol{x}_k^-$ directly. Furthermore, the code phase of the next epoch is calculated by $\boldsymbol{x}_{k+1}^- = \boldsymbol{x}_k^+ + \Delta\boldsymbol{x}_{k+1}^-$, which means the posterior user state increment of the previous epoch $\delta\boldsymbol{x}_k^+$ is included in the code phase predication of the current epoch. Hence, correction of the code phase as the relative VTL in Eq. (16) is not necessary in the absolute VTL.

From the above analysis, it is noted that both of the two VTLs adjust the code phase of the local replica to align with the received signals although in different manners, and the same navigation solution can be obtained in case the two VTL implementations are well initialized and the incoming signals are clean. Therefore, both of the two VTLs track the received signals normally and output correct navigation solutions in benign conditions.

3. Biases in VTLs

VTLs are commonly initialized by the traditional STL or other exterior sources.[14] VTLs' initialization includes the initial parameters for local replicas generation and initialization of the navigation filter. The initial user state bias and the initial code phase bias are discussed firstly in this section. Besides, the biases in VTLs' operational duration are also discussed, including erroneous navigation solution induced user state bias and multipath/NLOS reception induced tracking bias. The innovation point of this section is illustrating the self-robustness of the two VTLs to the above mentioned four biases systematically for the first time.

3.1. Initial user state bias

Let $\hat{\boldsymbol{x}}_0$ and $\hat{\boldsymbol{\tau}}_0$ denote the unbiased initial user state and the unbiased initial code phase, respectively, then these two terms satisfy the relation in Eq. (4), that is $\hat{\boldsymbol{\tau}}_0 = h(\hat{\boldsymbol{x}}_0)/c$.

The initial user state bias discussed here means that a bias $\boldsymbol{x}'_\text{b}$ appears in the initial user state, then the actual initial user state is $\hat{\boldsymbol{x}}'_0 = \hat{\boldsymbol{x}}_0 + \boldsymbol{x}'_\text{b}$. Nevertheless, the intial code phase is still the unbiased $\hat{\boldsymbol{\tau}}_0$, and hence $\hat{\boldsymbol{\tau}}_0 \neq h(\hat{\boldsymbol{x}}'_0)/c$. The quantities with a slash are used to represent the ones contaminated by initial user state bias. Considering the user state increment is not affected by initial user state bias, the user state prediction at the first epoch is

$$\boldsymbol{x}'^-_1 = \hat{\boldsymbol{x}}'_0 + \Delta\boldsymbol{x}^-_1 = \boldsymbol{x}^-_1 + \boldsymbol{x}'_\text{b} \tag{20}$$

which is also biased. According to Eq. (12), the code phase prediction at the first epoch of the relative VTL is represented as

$$\boldsymbol{\tau}^-_1 = \hat{\boldsymbol{\tau}}_0 + \Delta\boldsymbol{\tau}^-_1 = \hat{\boldsymbol{\tau}}_0 + \boldsymbol{H}_k\Delta\boldsymbol{x}^-_1/c = h(\boldsymbol{x}^-_1)/c \tag{21}$$

since the code phase predication $\boldsymbol{\tau}^-_1$ is controlled by the user state increment prediction, as the second equation in Eq. (21), it is not affected by the initial user state bias. It should be pointed out that the $\boldsymbol{x}^-_1$ in Eq. (21) is used only to reveal the relations among the quantities, which does not mean the code phase is calculated by $\boldsymbol{x}^-_1$. In the case of clean signal reception, which means the true parameters carried by the received signal are unbiased and the relation $\boldsymbol{\tau}_1 = h(\boldsymbol{x}_1)/c$ is established, the code phase residual is

$$\delta\boldsymbol{\tau}_1 = \boldsymbol{\tau}_1 - \boldsymbol{\tau}^-_1 = \boldsymbol{H}_k(\boldsymbol{x}_1 - \boldsymbol{x}^-_1)/c = \boldsymbol{H}_k\delta\boldsymbol{x}_1/c \tag{22}$$

Since the true code phase $\boldsymbol{\tau}_1$ and the prediction $\boldsymbol{\tau}^-_1$ are both unbiased, the measurement in Eq. (22) is not affected by the initial user state bias. Therefore, the unbiased user state residual estimation $\delta\hat{\boldsymbol{x}}_1$ can be obtained. By adding $\delta\hat{\boldsymbol{x}}_1$ to the predicted $\boldsymbol{x}'^-_1$, the navigation solution is given below

$$\boldsymbol{x}'^+_1 = \boldsymbol{x}'^-_1 + \delta\hat{\boldsymbol{x}}_1 = \boldsymbol{x}^+_1 + \boldsymbol{x}'_\text{b} \tag{23}$$

which is also biased for the biased initial user state. The primary reason of the biased navigation solution is that the code phase prediction $\boldsymbol{\tau}^-_1$ for the local replica in Eq. (21) is not directly coupled with the user state prediction $\boldsymbol{x}'^-_1$, and hence the bias term in $\boldsymbol{x}'^-_1$ is not included in the measurement in Eq. (22). On the other side, since $\delta\hat{\boldsymbol{x}}_1$ is unbiased, the code phase correction term obtained according to Eq. (15) is unbiased, then the posterior corrected code phase obtained by Eq. (16) is the unbiased one, i.e., $\boldsymbol{\tau}^+_1$. Therefore, the equation between the unbiased posterior code phase and the biased posterior user state is no longer established, i.e., $\boldsymbol{\tau}^+_1 \neq h(\boldsymbol{x}'^+_1)/c$.

The above discussion reveals that initial user state bias would not affect the tracking module of the relative VTL, and hence it cannot be eliminated by the new measurement at the first epoch. Actually, the initial user state bias cannot be eliminated in all the following epochs for the bias does not affect the new measurements in the following epochs. Therefore, the initial user state bias results in deviation from steady navigation solution in the following epochs of the relative VTL.

In the absolute VTL, the code phase prediction at the first epoch is

$$\boldsymbol{\tau}'^-_1 = g^{-1}\left(h\left(\boldsymbol{x}'^-_1\right)\right) \tag{24}$$

Since the initial user state bias is included in $\boldsymbol{x}'^-_1$, the code phase prediction $\boldsymbol{\tau}'^-_1$ is biased. The user state bias propagates to the code phase of the local replica, which is the backward bias propagation in VTLs. The code phase residual then becomes

$$\delta\boldsymbol{\tau}'_1 = \boldsymbol{\tau}_1 - \boldsymbol{\tau}'^-_1 = \boldsymbol{H}_k\left(\boldsymbol{x}_1 - \boldsymbol{x}'^-_1\right)/c \tag{25}$$

The propagated bias appears in the new measurement $\delta\boldsymbol{\tau}'_1$, which is coupled with the biased user state residual $\delta\boldsymbol{x}'_1 = \boldsymbol{x}_1 - \boldsymbol{x}'^-_1$. Therefore, the biased user state residual $\delta\hat{\boldsymbol{x}}'_1 = \delta\hat{\boldsymbol{x}}_1 - \hat{\boldsymbol{x}}'_\text{b}$ is estimated. By adding the estimation of biased user state residual to the biased user state prediction, we get the below navigation solution

$$\boldsymbol{x}'^+_1 = \boldsymbol{x}'^-_1 + \delta\hat{\boldsymbol{x}}'_1 = \boldsymbol{x}^+_1 \tag{26}$$

The second equation in Eq. (26) is established by ignoring the estimation error in $\delta\hat{\boldsymbol{x}}'_1$. In fact, the navigation solutions converge to the unbiased $\boldsymbol{x}^+_1$ gradually. The two equations in Eq. (26) reveal that the initial user state bias is eliminated by the new measurement in Eq. (25), and hence the navigation solution is unbiased.

A numerical simulation based on the semi-analytic model[9] is given to illustrate the response of the two VTLs as the initial user state bias appears. Assume there are 5 visible satellites with the azimuths = [210°, 30°, 120°, 180°, 247°] and elevations = [57°, 70°, 65°, 52°, 70°], and the C/N$_0$ of all the satellite signals are 43 dB-Hz. The initial value of the true user state is $\boldsymbol{x}_0 = [0, 0, 0, 0]$ m in the ECEF coordinate system, and the ground truth of the horizontal user trajectory are shown as the black dots in Fig. 1(a). The initial tracking parameters of the two VTLs are all identical to the true ones calculated by the user motion model.[9] Meanwhile, an initial user position bias of 60 m in the X and 10 m in the Y directions are added to the EKF's initial state of the two VTL implementations. The positioning results of the two VTLs are shown in Fig. 1 (a), while the horizontal positioning error against time is shown in Fig. 1(b). It can be seen from Fig. 1 that the initial bias still exists in the following navigation solutions of the relative VTL, whereas in the absolute VTL, the bias disappears after a short period for it converges to ground truth quickly.

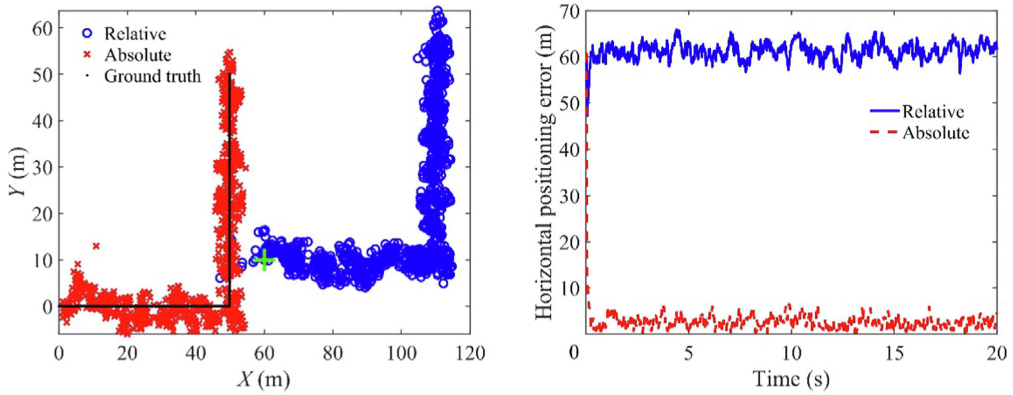
3.2. Initial tracking bias

The initial code phase bias here means a bias $\boldsymbol{\tau}''_\text{b}$ appears in the initial code phase for local code generation, that is to say, the initial code phase becomes $\hat{\boldsymbol{\tau}}''_0 = \hat{\boldsymbol{\tau}}_0 + \boldsymbol{\tau}''_\text{b}$, and the initial user state $\hat{\boldsymbol{x}}_0$ is unbiased, and hence $\hat{\boldsymbol{\tau}}''_0 \neq h(\hat{\boldsymbol{x}}_0)/c$.

Since the user state increment prediction is not affected by the initial code phase bias, the prior user state prediction $\boldsymbol{x}^-_1$ is unbiased. According to Eq. (12), the code phase prediction in the relative VTL is

$$\boldsymbol{\tau}''^-_1 = \hat{\boldsymbol{\tau}}''_0 + \boldsymbol{H}_k\Delta\boldsymbol{x}^-_1/c = \boldsymbol{\tau}^-_1 + \boldsymbol{\tau}''_\text{b} = h\left(\boldsymbol{x}''^-_1\right)/c \tag{27}$$

where $\boldsymbol{x}''^-_1 = \boldsymbol{x}^-_1 + c \cdot h^{-1}(\boldsymbol{\tau}''_\text{b}) = \boldsymbol{x}^-_1 + \boldsymbol{x}''_\text{b}$ stands for the biased user state prediction corresponding to the biased code phase prediction $\boldsymbol{\tau}''^-_1$, with the bias component $\boldsymbol{x}''_\text{b}$ originating from the biased code phase $\boldsymbol{\tau}''_\text{b}$. It should be pointed out that the



(a) Comparison of horizontal trajectories (b) Horizontal positioning error comparison

Fig. 1 Comparison of positioning results as initial user state bias exists.

actual user state prediction is the unbiased $\boldsymbol{x}^{-}_1$. As the true code phase corresponding to the clean received signal is $\boldsymbol{\tau}_1$, the code phase residual becomes

$$\delta\boldsymbol{\tau}''_1 = \boldsymbol{\tau}_1 - \boldsymbol{\tau}''^{-}_1 = \boldsymbol{H}_k\left(\boldsymbol{x}_1 - \boldsymbol{x}''^{-}_1\right)/c \tag{28}$$

The measurement in Eq. (28) is coupled with $\delta\boldsymbol{x}''_1 = \boldsymbol{x}_1 - \boldsymbol{x}''^{-}_1$, which leads to a biased user state residual estimation $\delta\hat{\boldsymbol{x}}''_1 = \delta\hat{\boldsymbol{x}}_1 - \hat{\boldsymbol{x}}''_{\mathrm{b}}$. The navigation solution is updated based on the actual user state prediction $\boldsymbol{x}^{-}_1$ and the biased user state residual estimation $\delta\hat{\boldsymbol{x}}''_1$ as below

$$\boldsymbol{x}''^{+}_1 = \boldsymbol{x}^{-}_1 + \delta\hat{\boldsymbol{x}}''_1 = \boldsymbol{x}^{+}_1 - \hat{\boldsymbol{x}}''_{\mathrm{b}} \tag{29}$$

It is noted that the initial code phase bias propagates to the navigation solution. By using $\delta\hat{\boldsymbol{x}}''_1$ to calculate the code phase correction term $\delta\hat{\boldsymbol{\tau}}''_1 = \delta\hat{\boldsymbol{\tau}}_1 - \boldsymbol{\tau}''_{\mathrm{bp}}$ according to Eq. (15), with $\boldsymbol{\tau}''_{\mathrm{bp}} = \boldsymbol{H}_k\hat{\boldsymbol{x}}''_b/c \neq \boldsymbol{\tau}''_{\mathrm{b}}$, the corrected code phase is

$$\boldsymbol{\tau}''^{+}_1 = \boldsymbol{\tau}''^{-}_1 + \delta\hat{\boldsymbol{\tau}}''_1 = \boldsymbol{\tau}^{+}_1 + \boldsymbol{\tau}''_{\mathrm{b}} - \boldsymbol{\tau}''_{\mathrm{bp}} \neq \boldsymbol{\tau}^{+}_1 \tag{30}$$

It is noted that an initial code phase bias also results in bias propagation among tracking channels, and then a combined bias term $\boldsymbol{\tau}''_{\mathrm{b}} - \boldsymbol{\tau}''_{\mathrm{bp}}$ will appear in the code phase of the local replicas. The navigation solution in Eq. (29) and the corrected code phase in Eq. (30) satisfy the equation $\boldsymbol{\tau}''^{+}_1 = h\left(\boldsymbol{x}''^{+}_1\right)/c$. Actually, the navigation solution bias as in Eq. (29) and the code phase bias in Eq. (30) exist in all the following epochs in the case of normal signal reception.

Based on the semi-analytic model in Ref. 9, the output of the code discriminator from the two VTLs are compared in Fig. 2, and the parameters of the satellites and the user are the same as the ones used in the results given in Section 3.1. In practical applications, the initial user state of VTLs is obtained from the navigation solution of STL with estimation error. To exclude the influence of initial user state bias, the EKF's initial user state of the two VTLs are assumed to be equal to the true one in this simulation. A 0.1 chip initial code phase bias is added to S1 to both of the two VTLs. It can be seen from Fig. 2(a) that other tracking channels of the relative VTL also show biases, while there is no bias in the tracking channels of the absolute VTL as shown in Fig. 2(b). This is in accordance with the analysis in this subsection. The 3-D positioning error and the clock bias estimation error are given in Fig. 3. It is noted that bias appears in the 3-D positioning error and the clock bias estimation error of the relative VTL, which means the initial code phase bias transfers to navigation solutions, and a steady bias almost exists in the navigation solutions of the relative VTL. However, the navigation solution of the absolute VTL converges to the unbiased true one quickly.

3.3. User state bias during VTLs' operation: Erroneous user state estimation

The EKF needs accurate statistical information about the user platform, which is usually estimated from the measurements in practical applications. As a user state jump happens,[34] accurate statistical information cannot be estimated immediately, and hence erroneous posterior user state would appear, i.e., a bias appears in the user state.

In the relative VTL, as a bias $\boldsymbol{x}'''_{\mathrm{b}}$ appears in the user state residual estimates $\delta\hat{\boldsymbol{x}}'''_k = \delta\hat{\boldsymbol{x}}_k + \boldsymbol{x}'''_{\mathrm{b}}$, the navigation solution is

$$\boldsymbol{x}'''^{+}_k = \boldsymbol{x}^{-}_k + \delta\hat{\boldsymbol{x}}'''_k = \boldsymbol{x}^{+}_k + \boldsymbol{x}'''_{\mathrm{b}} \tag{31}$$

According to Eq. (16), the corrected code phase is

$$\boldsymbol{\tau}'''^{+}_k = \boldsymbol{\tau}^{-}_k + \delta\hat{\boldsymbol{\tau}}'''_k = \boldsymbol{\tau}^{+}_k + \boldsymbol{H}_k\boldsymbol{x}'''_{\mathrm{b}}/c \tag{32}$$

Hence, the corrected code phase is biased for $\boldsymbol{H}_k\boldsymbol{x}'''_{\mathrm{b}}/c$. It is noted that the bias in the user state is transferred to a code phase bias of the local replica, which belongs to the backward bias propagation in VTLs. The relation between the corrected code phase $\boldsymbol{\tau}'''^{+}_k$ in Eq. (32) and the posterior user state $\boldsymbol{x}'''^{+}_k$ in Eq. (31) is still established, i.e., $\boldsymbol{\tau}'''^{+}_k = h\left(\boldsymbol{x}'''^{+}_k\right)/c$. In the case of a normal user state increment prediction, the user state prediction at the next epoch is

$$\boldsymbol{x}'''^{-}_{k+1} = \boldsymbol{x}'''^{+}_k + \Delta\boldsymbol{x}^{-}_{k+1} = \boldsymbol{x}^{-}_{k+1} + \boldsymbol{x}'''_{\mathrm{b}} \tag{33}$$

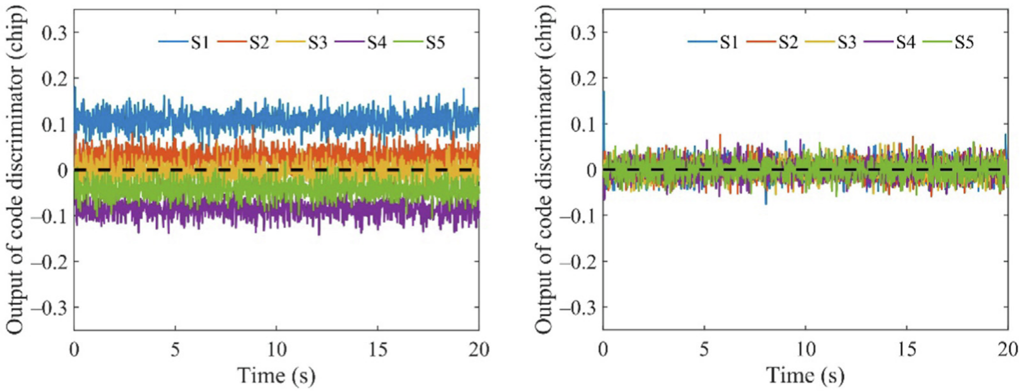
Then, the code phase prediction in the relative VTL is

$$\boldsymbol{\tau}'''^{-}_{k+1} = \boldsymbol{\tau}'''^{+}_k + \Delta\boldsymbol{\tau}^{-}_{k+1} = \boldsymbol{\tau}^{-}_{k+1} + \boldsymbol{H}_k\boldsymbol{x}'''_{\mathrm{b}}/c \tag{34}$$

with which the code phase residual can be represented as

$$\delta\boldsymbol{\tau}'''_{k+1} = \boldsymbol{\tau}_{k+1} - \boldsymbol{\tau}'''^{-}_{k+1} = \boldsymbol{H}_k\left(\boldsymbol{x}_{k+1} - \boldsymbol{x}'''^{-}_{k+1}\right)/c \tag{35}$$

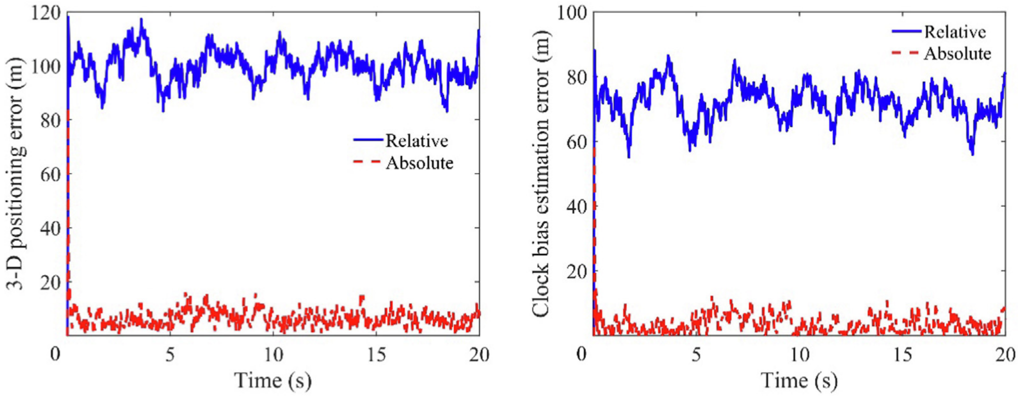
the measurement above is coupled with the user state residual $\delta\boldsymbol{x}'''_{k+1} = \boldsymbol{x}_{k+1} - \boldsymbol{x}'''^{-}_{k+1}$, and the estimation of the user state residual would compel the biased prediction $\boldsymbol{x}'''^{-}_{k+1}$ to converge



(a) Relative VTL

(b) Absolute VTL

Fig. 2 Comparison of output from code discriminator as initial code phase bias exists.



(a) 3-D positioning error

(b) Clock bias estimation error

Fig. 3 Comparison of navigation solution errors as initial code phase bias exists.

to the unbiased true user state $\boldsymbol{x}_{k+1}$. The estimated user state residual $\delta\hat{\boldsymbol{x}}'''_{k+1} = \delta\hat{\boldsymbol{x}}_{k+1} - \hat{\boldsymbol{x}}'''_{\mathrm{b}}$ contains a bias term $-\hat{\boldsymbol{x}}'''_{\mathrm{b}}$, which would eliminate the bias in Eq. (33) in the navigation solution at epoch $k + 1$ as below:

$$\boldsymbol{x}'''^{+}_{k+1} = \boldsymbol{x}'''^{-}_{k+1} + \delta\hat{\boldsymbol{x}}'''_{k+1} = \boldsymbol{x}^{+}_{k+1} \tag{36}$$

The last equation in Eq. (36) is established by ignoring the estimation error. The unbiased navigation solution can be seen from the final navigation solution in Eq. (36). Further, the corrected code phase by $\delta\hat{\boldsymbol{x}}'''_{k+1}$ according to Eqs. (15) and (16) is also unbiased as

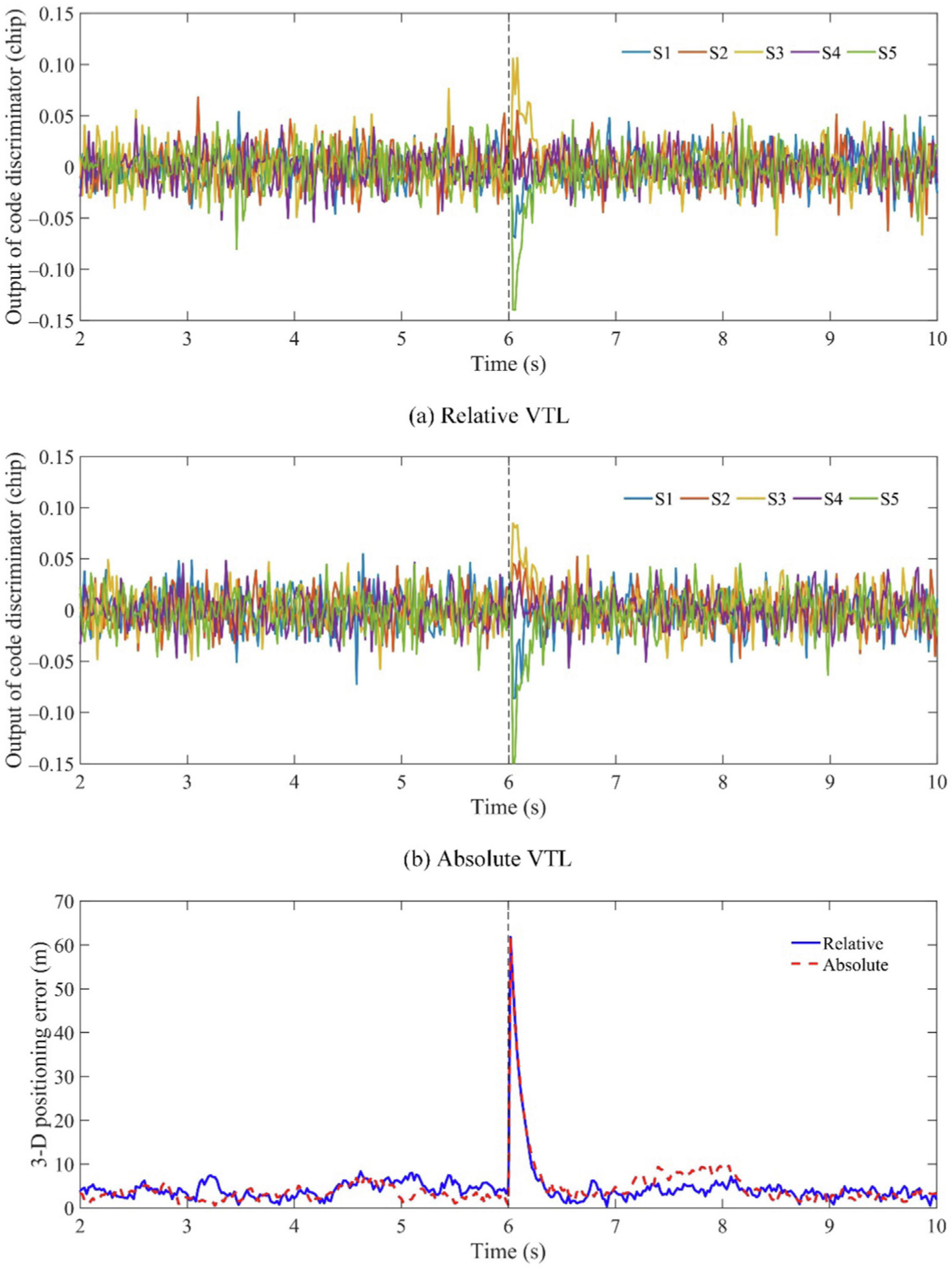
$$\boldsymbol{\tau}'''^{+}_{k+1} = \boldsymbol{\tau}'''^{-}_{k+1} + \boldsymbol{H}_k\delta\hat{\boldsymbol{x}}'''_{k+1}/c = \boldsymbol{\tau}^{+}_{k+1} \tag{37}$$

Therefore, we can conclude that a user state residual estimation bias can be eliminated by the new measurements.

Considering a bias appears in the user state increment prediction, the bias appears in the code phase increment prediction and further appears in the code phase prediction of the next epoch, which will result in similar phenomenon as the above discussed user state residual estimation bias. Both of the above two user state bias would result in a code phase bias prediction at the next epoch, and the relation between the code phase and the user state is still established. Hence, in the case of normal signal reception, the navigation solution and the code phase of the local replica would be dragged to the unbiased ones.

For the absolute VTL, since both of the above two user state biases appear in the user state prediction of the next epoch $\boldsymbol{x}'''^{-}_{k+1}$, the code phase prediction $\boldsymbol{\tau}'''^{-}_{k+1}$ according to Eq. (17) is biased. Hence, the same measurement as that in Eq. (35) can be obtained, and thereby the same unbiased navigation solution can be obtained. Hence, the user state biases during the VTL operational period do not affect the performance of the absolute VTL in the following epochs.

A simulation based on the semi-analytic model[9] is given to verify the above analysis. The parameters of the satellites and the user are all the same as the ones used in the results given Section 3.1. The initial tracking parameters and the initial user state are all identical to the ground truth, while a user position bias [50, 10, 0] m is added at 6 s to both of the two VTLs. The output of the code discriminator from the two VTL implementations and the 3-D positioning error are shown in Fig. 4. It can be seen from the figure that tracking channels and the navigation solution of the two VTLs converge to the unbiased true ones quickly. Actually, the clock biases of the two VTLs are not affected by the added bias, which are not shown in Fig. 4.



(a) Relative VTL

(b) Absolute VTL

(c) Comparison of 3-D positioning error

Fig. 4 Comparison of results as a user state bias appears at 6 s.

3.4. Tracking bias during VTLs' operation: Abnormal signal reception

Assume the multipath/NLOS induced tracking bias appears at epoch $k-1$ and then disappears at epoch k, its effect on the two VTLs are discussed in this subsection. As multipath/NLOS appears, a nominal code phase vector of the received signals is defined as $\boldsymbol{\tau}''''_{k-1} = \boldsymbol{\tau}_{k-1} + \boldsymbol{\tau}''''_{k-1,\mathrm{b}}$, which is the code phase corresponding to the combined signal for the multipath or the code phase of the reflected signal for the NLOS reception. The term $\boldsymbol{\tau}''''_{k-1,\mathrm{b}}$ denotes the multipath/NLOS-induced code phase bias, and the zero-value elements of $\boldsymbol{\tau}''''_{k-1,\mathrm{b}}$ denote the signals without bias. Based on the relations in Eq. (4), the true user state $\boldsymbol{x}''''_{k-1} = \boldsymbol{x}_{k-1} + \boldsymbol{x}''''_{k-1,\mathrm{b}}$ from the signal is biased, with $\boldsymbol{x}''''_{k-1,\mathrm{b}} = c \cdot h^{-1}\left(\boldsymbol{\tau}''''_{k-1,\mathrm{b}}\right)$. As the multipath/NLOS disappears at epoch k, the true unbiased $\boldsymbol{x}_k$ and $\boldsymbol{\tau}_k$ corresponding to clean signals recover.

Based on the normal navigation solution at the previous epoch, unbiased normal prediction $\Delta\boldsymbol{x}^{-}_{k-1}$ and $\boldsymbol{x}^{-}_{k-1}$ can be obtained, which further leads to unbiased $\Delta\boldsymbol{\tau}^{-}_{k-1}$ and $\boldsymbol{\tau}^{-}_{k-1}$. Therefore, the code phase residuals of the two VTLs at epoch $k-1$ are the same as

$$\delta\boldsymbol{\tau}''''_{k-1} = \delta\boldsymbol{\tau}_{k-1} + \boldsymbol{\tau}''''_{k-1,\mathrm{b}} = \boldsymbol{H}_k\left(\boldsymbol{x}''''_{k-1} - \boldsymbol{x}^-_{k-1}\right)/c \tag{38}$$

The measurement in Eq. (38) is coupled with the biased user state residual $\delta\boldsymbol{x}''''_{k-1} = \boldsymbol{x}''''_{k-1} - \boldsymbol{x}^-_{k-1}$. The user state residual estimation can be represented as $\delta\hat{\boldsymbol{x}}''''_{k-1} = \delta\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{x}}''''_{k-1,\mathrm{b}}$, where the biased user state term $\hat{\boldsymbol{x}}''''_{k-1,\mathrm{b}}$ is originated from multipath/NLOS-induced code phase bias $\boldsymbol{\tau}''''_{k-1,\mathrm{b}}$. Further, the navigation solution according to Eq. (10) is

$$\boldsymbol{x}''''^{+}_{k-1} = \boldsymbol{x}^-_{k-1} + \delta\hat{\boldsymbol{x}''''}_{k-1} = \boldsymbol{x}^+_{k-1} + \hat{\boldsymbol{x}}''''_{k-1,\mathrm{b}} \tag{39}$$

The above navigation solution is the same for both of the two VTL implementations. With $\delta\hat{\boldsymbol{x}}''''_{k-1}$, the corrected code phase of the relative VTL can be obtained according to Eq. (16) as

$$\boldsymbol{\tau}''''^{+}_{k-1} = \boldsymbol{\tau}^-_{k-1} + \delta\hat{\boldsymbol{\tau}''''}_{k-1} = \boldsymbol{\tau}^+_{k-1} + \boldsymbol{\tau}_{k-1,\mathrm{bp}} \tag{40}$$

where $\boldsymbol{\tau}_{k-1,\mathrm{bp}} = \boldsymbol{H}_k\boldsymbol{x}''''_{k-1,\mathrm{b}}/c \neq \boldsymbol{\tau}''''_{k-1,\mathrm{b}}$, and hence the bias in the received signals is transferred to the code phase of the local replica, which is the backward bias propagation in VTLs. The relation between $\boldsymbol{x}''''^{+}_{k-1}$ and $\boldsymbol{\tau}''''^{+}_{k-1}$ is still established, i.e., $\boldsymbol{\tau}''''^{+}_{k-1} = h(\boldsymbol{x}''''^{+}_{k-1})/c$.

The user state increment prediction of the next epoch is $\Delta\boldsymbol{x}^-_k$, and hence the user state prediction is

$$\boldsymbol{x}''''^{-}_{k} = \boldsymbol{x}''''^{+}_{k-1} + \Delta\boldsymbol{x}^-_k \tag{41}$$

It is noted from Eq. (41) that the code phase prediction of the two VTLs would be the same, which is represented as

$$\boldsymbol{\tau}''''^{-}_{k} = \boldsymbol{\tau}^-_k + \boldsymbol{\tau}_{k-1,\mathrm{bp}} \tag{42}$$

As the signal recovers, i.e., the multipath/NLOS disappears at epoch k, the code phase residual is represented as

$$\delta\boldsymbol{\tau}''''_{k} = \boldsymbol{\tau}_k - \boldsymbol{\tau}''''^{-}_{k} = \boldsymbol{H}_k\delta\left(\boldsymbol{x}_k - \boldsymbol{x}''''^{-}_{k}\right)/c \tag{43}$$

The biased user state residual $\delta\hat{\boldsymbol{x}}''''_k = \delta\hat{\boldsymbol{x}}_k - \hat{\boldsymbol{x}}''''_{k-1,\mathrm{b}}$ can be estimated from the above measurement. Further, the navigation solution is updated as

$$\boldsymbol{x}''''^{+}_{k} = \boldsymbol{x}''''^{-}_{k-1} + \delta\hat{\boldsymbol{x}''''}_{k} = \boldsymbol{x}^+_k \tag{44}$$

and the last equation in Eq. (44) is established by ignoring the estimation error. It is seen from Eq. (44) that unbiased navigation solution is obtained, which can also be illustrated by the measurement in Eq. (43), where the measurement is coupled with the difference between the unbiased true user state $\boldsymbol{x}_k$ and the biased user state prediction $\boldsymbol{x}''''^{-}_{k}$. Therefore, the estimated user state residual would compel $\boldsymbol{x}''''^{-}_{k}$ to converge to $\boldsymbol{x}_k$. The posterior corrected code phase becomes

$$\boldsymbol{\tau}''''^{+}_{k} = \boldsymbol{\tau}''''^{-}_{k} + \boldsymbol{H}_k\delta\hat{\boldsymbol{x}}''''_k/c = \boldsymbol{\tau}^+_k \tag{45}$$

It is noted that the new measurement compels the biased code phase prediction $\boldsymbol{\tau}''''^{-}_{k}$ to align with the unbiased true one $\boldsymbol{\tau}_k$; meanwhile, the biased user state prediction $\boldsymbol{x}''''^{-}_{k}$ is converged to the unbiased true one $\boldsymbol{x}_k$.

Based on the above analysis, it is noted that the multipath/NLOS induced tracking bias and in turn navigation solution bias disappear as the signal recovers, which is true for both of the two VTLs. In fact, the tracking loops and the navigation filter are adjusted by the clean LOS signal as the signal recovers, which compels the navigation solution converges to the unbiased true one.

A simulation based on the semi-analytic model[9] is given to verify the above analysis of multipath/NLOS induced tracking bias. The parameters of the satellites and the user are all the same as those in Section 3.1. No user state bias and tracking bias are added, while an NLOS reception signal with an extra code phase of 0.25 chips is added to S1 during 8–12 s. The output of the code discriminator from the relative VTL and the absolute VTL are given in Fig. 5(a) and Fig. 5(b), respectively. It can be seen that the NLOS induced tracking bias propagates to all the other tracking channels. The 3-D positioning error and the clock bias estimation error from the two VTLs are compared in Fig. 5(c) and Fig. 5(d). The comparison shows that the two VTLs exhibit similar navigation solution bias during the NLOS period, while the navigation solution converges to the unbiased one as the NLOS disappears.

Based on the analysis in Section 3.1-3.4, robustness of the two VTLs to the above mentioned four biases are concluded in Table 1.

3.5. *Key factors for tracking bias correction*

Since the tracking bias affects the two VTLs in the likely manner as discussed in Section 3.4, only the absolute VTL is taken as an example to illustrate the tracking bias in the duration of multipath/NLOS. To deal with the multipath/NLOS effectively, the key factor that affects bias correction performance is discussed in this section.

Considering the multipath/NLOS still exists at epoch k, the nominal true code phase is $\boldsymbol{\tau}''''_k = \boldsymbol{\tau}_k + \boldsymbol{\tau}_{k,\mathrm{b}}$, with $\boldsymbol{\tau}_{k,\mathrm{b}}$ denoting the multipath/NLOS induced tracking bias at the current epoch; thereby, the code phase residual becomes

$$\begin{aligned}\delta\boldsymbol{\tau}''''_k &= \boldsymbol{\tau}''''_k - \boldsymbol{\tau}''''^{-}_k = \delta\boldsymbol{\tau}_k + \left(\boldsymbol{\tau}_{k,\mathrm{b}} - \boldsymbol{\tau}_{k-1,\mathrm{bp}}\right)\\ &= \boldsymbol{H}_k\left(\boldsymbol{x}''''_k - \boldsymbol{x}''''^{-}_k\right)/c\end{aligned} \tag{46}$$

From the term in the first bracket, it is noted that the measurement residuals of all the tracking channels show bias for the propagated tracking bias term $-\boldsymbol{\tau}_{k-1,\mathrm{bp}}$. The measurement in Eq. (46) is coupled with the user state residual $\delta\boldsymbol{x}''''_k = \boldsymbol{x}''''_k - \boldsymbol{x}''''^{-}_k$, with which the user state residual estimation is $\delta\hat{\boldsymbol{x}}''''_k = \delta\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{x}}_{k,\mathrm{b}} - \hat{\boldsymbol{x}}_{k-1,\mathrm{b}}$; hence, the navigation solution can be updated

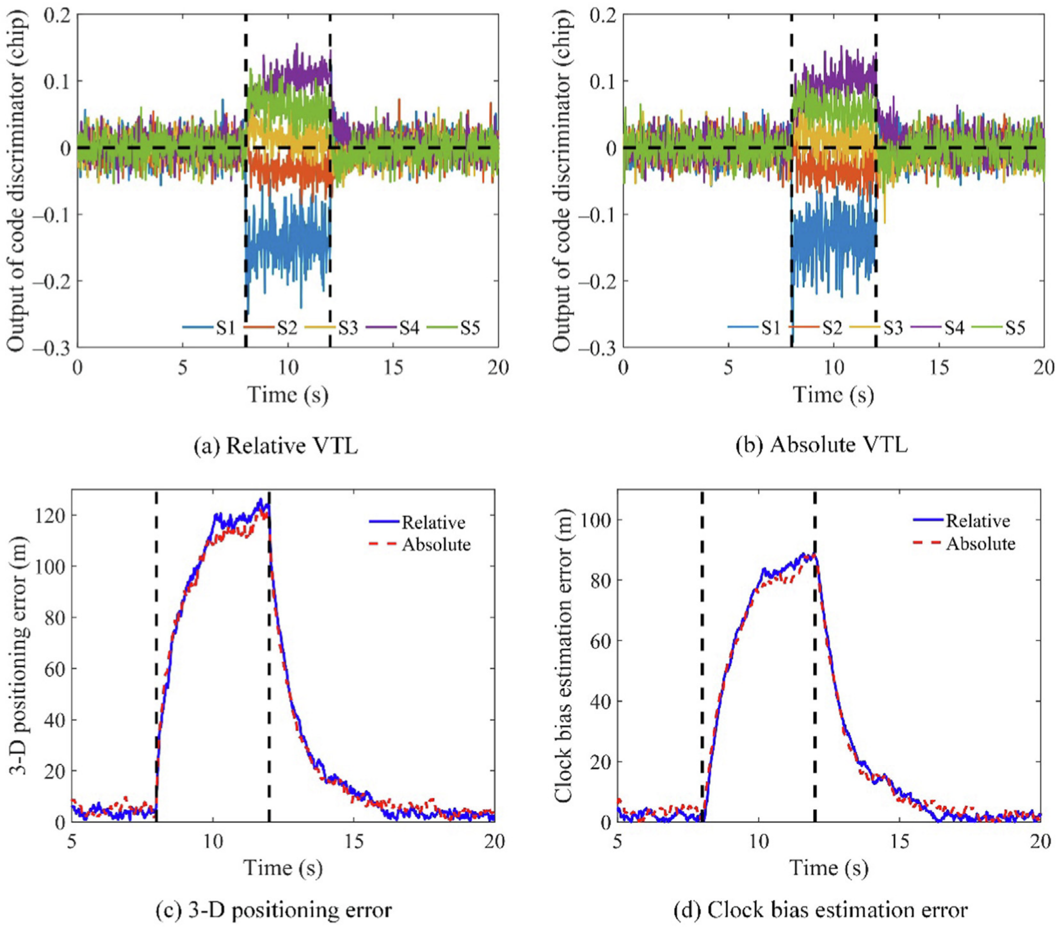
$$\boldsymbol{x}''''^{+}_k = \boldsymbol{x}''''^{-}_k + \delta\hat{\boldsymbol{x}}''''_k = \boldsymbol{x}^+_k + \boldsymbol{x}^+_{k,\mathrm{b}} \tag{47}$$

The bias in Eq. (47) is only related with the multipath/NLOS induced tracking bias of the current epoch.

To eliminate the navigation bias when the multipath/NLOS exists, the tracking bias in Eq. (46) is estimated and corrected.[10] Suppose the estimated bias from the discriminator's output is equal to the true bias in Eq. (46) for simplicity, then the corrected code phase residual is equal to $\delta\boldsymbol{\tau}_k$, which leads to the posterior user state residual $\delta\hat{\boldsymbol{x}}_k$. Finally, the navigation solution would be

$$\breve{\boldsymbol{x}}''''^{+}_{k} = \breve{\boldsymbol{x}}''''^{-}_{k} + \delta\hat{\boldsymbol{x}}_k = \boldsymbol{x}^+_k + \boldsymbol{x}^+_{k-1,\mathrm{b}} \tag{48}$$

which is still biased for the propagated tracking bias from the previous epoch. Therefore, to eliminate the tracking bias completely, it is not the tracking bias with the propagated term in Eq. (46), but the multipath/NLOS introduced tracking bias at each epoch should be corrected. However, it is not easy to separate the two bias terms in Eq. (40). Actually, the tracking bias in Eq. (38) denotes the first epoch that the multipath/NLOS appears, which contains only the multipath/NLOS introduced



(a) Relative VTL

(b) Absolute VTL

(c) 3-D positioning error

(d) Clock bias estimation error

Fig. 5 Comparison of tracking and navigation solution as a NLOS appears during 8–12 s.

Table 1 Comparison between the two VTLs.

VTL type Category		Relative VTL		Absolute VTL
		Tracking channel	Navigation solution	
Initial bias	User state	Does not affect the tracking channels	Bias exists in navigation solutions of the following epochs	Can be eliminated by the new measurements
	Code phase	Results in bias propagation among tracking channels	Transfer to navigation solution bias that exists in the following navigation solutions	Can be eliminated by the new measurements
Bias during VTLs	User state	Can be eliminated by the new measurements		
	Tracking	Induces navigation bias and tacking bias propagation, which disappear as the signal recovers		

tracking bias at the current epoch. Hence, it is important to catch sight of the tacking bias immediately as it appears, and further correct it timely at each epoch before its propagation.

4. Proposed robust VTL structure

Considering the absolute VTL is robust to initialization biases, it is selected in the proposed robust VTL structure. To further eliminate the multipath/NLOS induced bias in its duration, an instant tracking bias detection and correction method is used. The flow chart of the proposed VTL structure with tracking bias correction is shown in Fig. 6, and the pseudorange residuals are used to calculate firstly a global test statistic to detect the potential of a tracking bias. A local test is turned on when the global test statistic exceeds the threshold. The local test is used to identify the exact biased channel. The bias in the measurement is estimated and corrected afterwards.

4.1. Bias detection

To simplify the expression, we take the k-th epoch as an example to illustrate the problem. Besides, we also omit the slashes in the superscript, since we have no information about the appearance of the multipath/NLOS before bias detection.

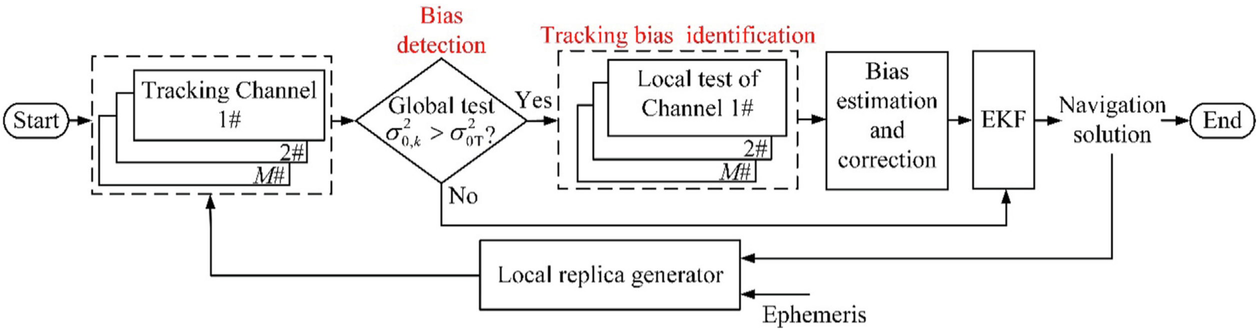


Fig. 6 Block diagram of the proposed VTL structure with tracking bias correction.

Based on the fact that the multipath/NLOS induced bias propagates to the pseudorange observations, a binary hypothesis is made below

$$\mathrm{H}_0 : \boldsymbol{\rho}_k = h(\boldsymbol{x}_k) + \boldsymbol{\varepsilon}_k \text{ and } \mathrm{H}_\mathrm{a} : \boldsymbol{\rho}_k = h(\boldsymbol{x}_k) + \boldsymbol{\rho}_{k,\mathrm{b}} + \boldsymbol{\varepsilon}_k \tag{49}$$

where $\boldsymbol{\rho}_{k,\mathrm{b}}$ denotes the multipath/NLOS induced pseudorange bias at the epoch k; The null-hypothesis H_0 stands for the case free of multipath/NLOS, while H_a denotes the alternative one with multipath/NLOS induced tracking bias. Assuming a normal user state prediction $\boldsymbol{x}_k^-$ can be obtained, then we re-express the pseudorange residual in Eq. (8) as

$$\delta\boldsymbol{\rho}_k = \boldsymbol{\rho}_k - h\left(\boldsymbol{x}_k^-\right) \tag{50}$$

under the null-hypothesis H_0, the pseudorange residual is Gaussian distributed with zero mean and the covariance matrix $\boldsymbol{R}_{\delta\rho_k} = \boldsymbol{R}_{\varepsilon_k} + \boldsymbol{H}_k\boldsymbol{P}_k^-\boldsymbol{H}_k^\mathrm{T}$, which does not hold when a bias appears. Hence, the binary hypothesis in Eq. (49) is equivalent to the one below

$$\mathrm{H}_0 : \mathrm{E}\{\delta\boldsymbol{\rho}_k\} = 0 \text{ and } \mathrm{H}_\mathrm{a} : \mathrm{E}\{\delta\boldsymbol{\rho}_k\} = \boldsymbol{\rho}_{k,\mathrm{b}} \tag{51}$$

It is known that as $\delta\boldsymbol{\rho}_k$ is Gaussian, its variance obeys chi square (χ^2) distribution. A global test statistic for detecting an inconsistent pseudorange residuals is defined as[3–4]

$$\sigma_{0,k}^2 = \frac{(\delta\boldsymbol{\rho}_k)^\mathrm{T}\boldsymbol{R}_{\delta\rho_k}^{-1}\delta\boldsymbol{\rho}_k}{M-4} \tag{52}$$

The global test is to verify whether $\sigma_{0,k}^2$ is centrally χ^2 distributed. $M-4$ is the degree of freedom. For the number of the available measurements is M, the unknown parameters in the user state $\boldsymbol{x}_k$ is 4. In the global test, the null-hypothesis H_0 states that the distributional assumptions meet the reality, as opposed to the alternative H_a, which states that the distributional assumptions are not correct. The threshold for the global test, $\sigma_{0\mathrm{T}}^2$, is defined as

$$\sigma_{0\mathrm{T}}^2 = \frac{\chi_{1-\alpha,M-4}^2}{M-4} \tag{53}$$

where α represents the false alarm rate. If the test statistic $\sigma_{0,k}^2$ exceeds the threshold $\sigma_{0\mathrm{T}}^2$, an inconsistency in the observations is assumed, in that way the biased channel should be identified.

The local test is carried out for bias channel identification, which is started only if a potential bias is detected in the global test. The local test statistic is defined as the standardized residual below[2]

$$w_k^m = \left|\frac{\delta\rho_k^m}{\sqrt{\boldsymbol{R}_{\delta\rho_k}^m}}\right|, \ m = 1, 2, \ldots, M \tag{54}$$

where $\boldsymbol{R}_{\delta\rho_k}^m$ denotes the m-th diagonal element of the covariance matrix $\boldsymbol{R}_{\delta\rho_k}$. Each standardized residual w_k^m is compared with the α_0-quantile of the standard normal distribution $n_{1-\alpha_0/2}$[3–4]

$$w_k^m \leqslant n_{1-\alpha_0/2}, \ m = 1, 2, \ldots, M \tag{55}$$

with a predetermined false alarm rate α_0. The null-hypothesis $\mathrm{H}_{0,m}$ denotes that the m-th observation is not an outlier, and is rejected if w_k^m exceeds the threshold. The underlying assumptions of this local test include that the measurement residuals in Eq. (54) follows $N\left(0, \boldsymbol{R}_{\delta\rho_k}\right)$ distribution is correct except for the single constant bias appears in the m-th observation. The standardized residual is then normally distributed with zero expectation when $\mathrm{H}_{0,m}$ is correct, and with a non-zero expectation otherwise.

4.2. Bias estimation and correction

A LASSO based method is applied to estimate the multipath/NLOS induced tracking bias for its correction. This method was firstly proposed for multipath induced tracking bias estimation in STLs.[33] To discuss the bias estimation, the pseudorange residual in Eq. (50) is formulated as

$$\delta\boldsymbol{\rho}_k = \boldsymbol{H}_k\delta\boldsymbol{x}_k + \boldsymbol{\rho}_{k,\mathrm{b}} + \boldsymbol{\varepsilon}_k \tag{56}$$

The unknowns in Eq. (56) are $\delta\boldsymbol{x}_k$ and $\boldsymbol{\rho}_{k,\mathrm{b}}$. Generally, since the number of multipath or NLOS affected channels is much smaller than the number of the in-view satellites, it is reasonable to assume that the bias vector $\boldsymbol{\rho}_{k,\mathrm{b}}$ is sparse. Exploiting this sparsity assumption, the user state and the measurement bias estimation can be obtained by solving the cost function below[33]

$$\delta\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{\rho}}_{k,\mathrm{b}} = \underset{\delta\boldsymbol{x}_k, \boldsymbol{\rho}_{k,\mathrm{b}}}{\arg\min} \| \delta\boldsymbol{\rho}_k - \boldsymbol{H}_k\delta\boldsymbol{x}_k - \boldsymbol{\rho}_{k,\mathrm{b}} \|_2^2 + \| \boldsymbol{W}_k\boldsymbol{\rho}_{k,\mathrm{b}} \|_1 \tag{57}$$

where $\boldsymbol{W}_k$ is a diagonal weighting matrix, and $\| \boldsymbol{W}_k\boldsymbol{\rho}_{k,\mathrm{b}} \|_1$ is the regulation term. Ideally, the weights contained in $\boldsymbol{W}_k$ should be inversely proportional to the magnitude of the unknown bias vector $\boldsymbol{\rho}_{k,\mathrm{b}}$.[33] For a fixed bias $\boldsymbol{\rho}_{k,\mathrm{b}}$, the least square solution of the user state residual is given by

$$\delta\widehat{\boldsymbol{x}}_k = \left(\boldsymbol{H}_k^{\mathrm{T}}\boldsymbol{H}_k\right)^{-1}\boldsymbol{H}_k^{\mathrm{T}}\left(\delta\boldsymbol{\rho}_k - \boldsymbol{\rho}_{k,\mathrm{b}}\right) \tag{58}$$

Utilizing the estimation $\delta\hat{\boldsymbol{x}}_k$ from Eq. (58) to replace the unknown $\delta\boldsymbol{x}_k$ in Eq. (57), the bias can be obtained by minimizing the cost function below

$$\hat{\boldsymbol{\rho}}_{k,\mathrm{b}} = \arg\min_{\boldsymbol{\rho}_{k,\mathrm{b}}} \frac{1}{2}\| \boldsymbol{P}_{\mathrm{H}}\left(\delta\boldsymbol{\rho}_k - \boldsymbol{\rho}_{k,\mathrm{b}}\right) \|_2^2 + \| \boldsymbol{W}_k\boldsymbol{\rho}_{k,\mathrm{b}} \|_1 \tag{59}$$

where $\boldsymbol{P}_{\mathrm{H}} = \boldsymbol{I} - \boldsymbol{H}_k\left(\boldsymbol{H}_k^{\mathrm{T}}\boldsymbol{H}_k\right)^{-1}\boldsymbol{H}_k^{\mathrm{T}}$ is the orthogonal projection matrix of the space spanned by the column vector of $\boldsymbol{H}_k$. It is noted that bias vector estimation is obtained by projecting the bias-corrected measurements $\delta\boldsymbol{\rho}_k - \boldsymbol{\rho}_{k,\mathrm{b}}$ onto the orthogonal space of the one spanned by the vector of $\boldsymbol{H}_k$.

Define

$$\begin{cases} \delta\bar{\boldsymbol{\rho}}_k = \boldsymbol{P}_{\mathrm{H}}\delta\boldsymbol{\rho}_k \\ \bar{\boldsymbol{H}}_k = \boldsymbol{P}_{\mathrm{H}}\boldsymbol{W}_k^{-1} \\ \boldsymbol{\theta}_k = \boldsymbol{W}_k\boldsymbol{\rho}_{k,\mathrm{b}} \end{cases} \tag{60}$$

Insert Eq. (60) into (59), we can get the following LASSO estimation problem

$$\hat{\boldsymbol{\rho}}_{k,b} = \arg\min_{\boldsymbol{\rho}_{k,\mathrm{b}}} \frac{1}{2}\| \delta\bar{\boldsymbol{\rho}}_{k,b} - \bar{\boldsymbol{H}}_k\boldsymbol{\theta}_k \|_2^2 + \lambda_k\| \boldsymbol{\theta}_k \|_1 \tag{61}$$

Then, the LASSO problem can be solved by using classical efficient algorithms. After the bias is estimated, the EKF measurement is corrected by

$$\delta\breve{\boldsymbol{\rho}}_k = \delta\boldsymbol{\rho}_k - \hat{\boldsymbol{\rho}}_{k,\mathrm{b}} \tag{62}$$

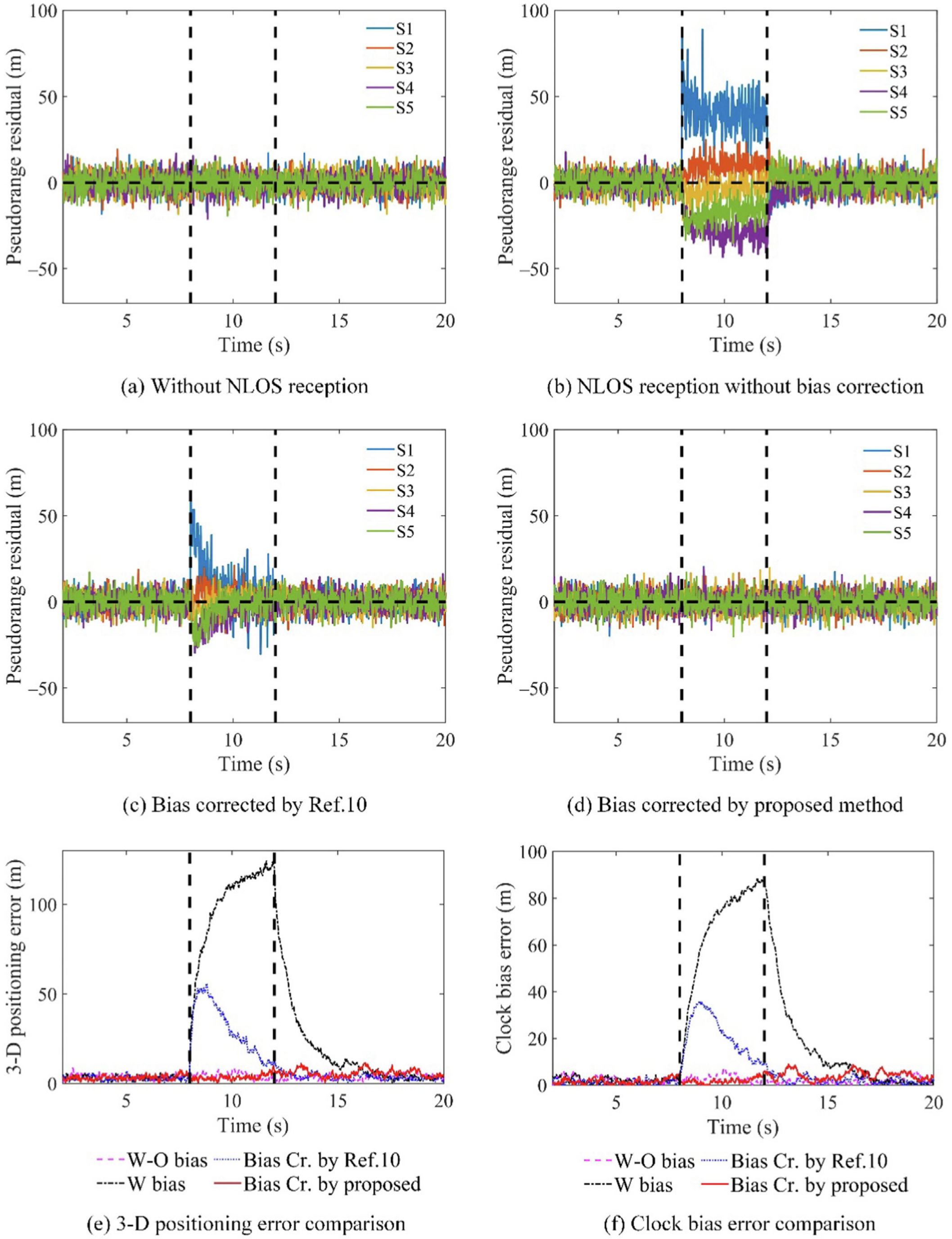
Since the multipath/NLOS- induced tracking bias is corrected before being sent to the EKF, the tracking bias induced navigation solution bias and the in-turn tracking bias propagation would not appear.

5. Results of proposed robust VTL

Effectiveness of the method to eliminate the multipath/NLOS induced tracking bias is verified by numerical simulations based on a semi-analytic model[9] and a field data test processed with the Software Defined Receiver (SDR). In the semi-analytic simulation, the constant velocity model is used to generate the user trajectory. There are five satellites with their elevation and azimuth are the same as given in section 3.1. The EKF update interval in the numerical simulations is 20 ms. Details of the process noise can be found in Ref. 1, and the measurement noise is randomly generated white Gaussian process. The C/N$_0$ of all the satellite signals are all 43 dB-Hz. An NLOS is simulated in S1 during 8-12 s, the extra delay of the NLOS is 0.25 chips relative to the LOS signal, and the amplitude of it is half of the LOS one, while the carrier frequency remains the same with that of the LOS signal. The pseudorange residuals are compared in Fig. 7. It is noted that from the case free of multipath/NLOS as shown in Fig. 7(a), the pseudorange residuals of all the tracking channels are zero mean. From the NLOS case without bias correction in Fig. 7(b), it is noted that the NLOS channel shows obvious bias, and the measurements from the other channels almost all deviate from zero mean. The above phenomenon is originated from the tracking bias propagation in VTLs. Fig. 7(c) shows the pseudorange residuals after bias correction by the method given in Ref. 10, where the bias is corrected based on the statistical mean of the measurements. This means that the bias is corrected after its propagation; hence, the bias cannot be corrected fully. It is clearly noted that the measurements of almost all the channels still show a slight bias. The pseudorange residuals after bias correction by the proposed method is shown in Fig. 7(d), where the means of all the measurements are zero for the bias of the NLOS channel is effectively corrected. The 3-D positioning error and the clock bias estimation error in different cases are compared, which are shown respectively in Figs. 7(e) and (f). It can be seen that the proposed method shows best positioning and timing performance, which are close to the ones without NLOS reception induced tracking bias.

Fig. 8 gives the statistic results in the situations of NLOS, constructive multipath, and destructive multipath. The relative code delay between the reflected signal and the LOS signal varies from 0 to 1.5 chips, while the amplitude of the reflected signal is half of the LOS one. The results of the NLOS case are given on the top two panels, while the results of a constructive multipath test case are given on the middle two panels, and the destructive multipath is shown on the bottom of the two panels. The mean of the measurements from the multipath/NLOS channel during the multipath/NLOS period is shown on the left panels, and is obtained by calculating the mean of the measurement from the multipath/NLOS channel in its duration. While the mean of the 3-D positioning error is shown on the right panels, and is obtained by calculating the mean of the 3-D positioning error in multipath/NLOS duration. All the results are obtained by Monte Carlo simulations with 50 trails. It can be seen clearly from Fig. 8 that the measurement after bias correction with the proposed method is close to zero, which is much superior to the results processed by the previously proposed bias correction method. On the other side, the mean of the 3-D positioning error of the proposed bias correction method is close to the one without NLOS reception, which further verified the effectiveness of the proposed bias correction method.

To further verify the performance of the proposed robust VTL, the GPS data collected from an open sky scenario is used in this experiment, and the detailed description of the data can be found in Ref. 10. A semi-physic experiment is carried out by manually adding an NLOS reception to PRN10 during 50–53.8 s to the field collected data. The extra delay of the NLOS signal is 0.5 chips, and its relative amplitude to the canceled directed signal is 0.5. The tracking results without and with bias correction by the method proposed in Ref. 10, and the results after bias correction by the proposed method are shown on the top three panels, while the horizontal positioning performance are compared on the bottom panel of Fig. 9. It can be seen that without tracking bias correction, the NLOS induced tracking bias to PRN 10 channel has propagated to all the other tracking channels. From Fig. 9(b), it is noted that the phenomenon of bias propagation is eliminated to a large extent after bias correction; however, there still remains some tracking bias propagation at the beginning as the NLOS appears. While from Fig. 9(c), it is noted that there is almost no bias propagation for the NLOS induced tracking bias can be corrected by the proposed method instantly. We define a total propagated tracking bias to evaluate the bias propagation phenomenon in the relative VTL. The total propagated



(a) Without NLOS reception

(b) NLOS reception without bias correction

(c) Bias corrected by Ref.10

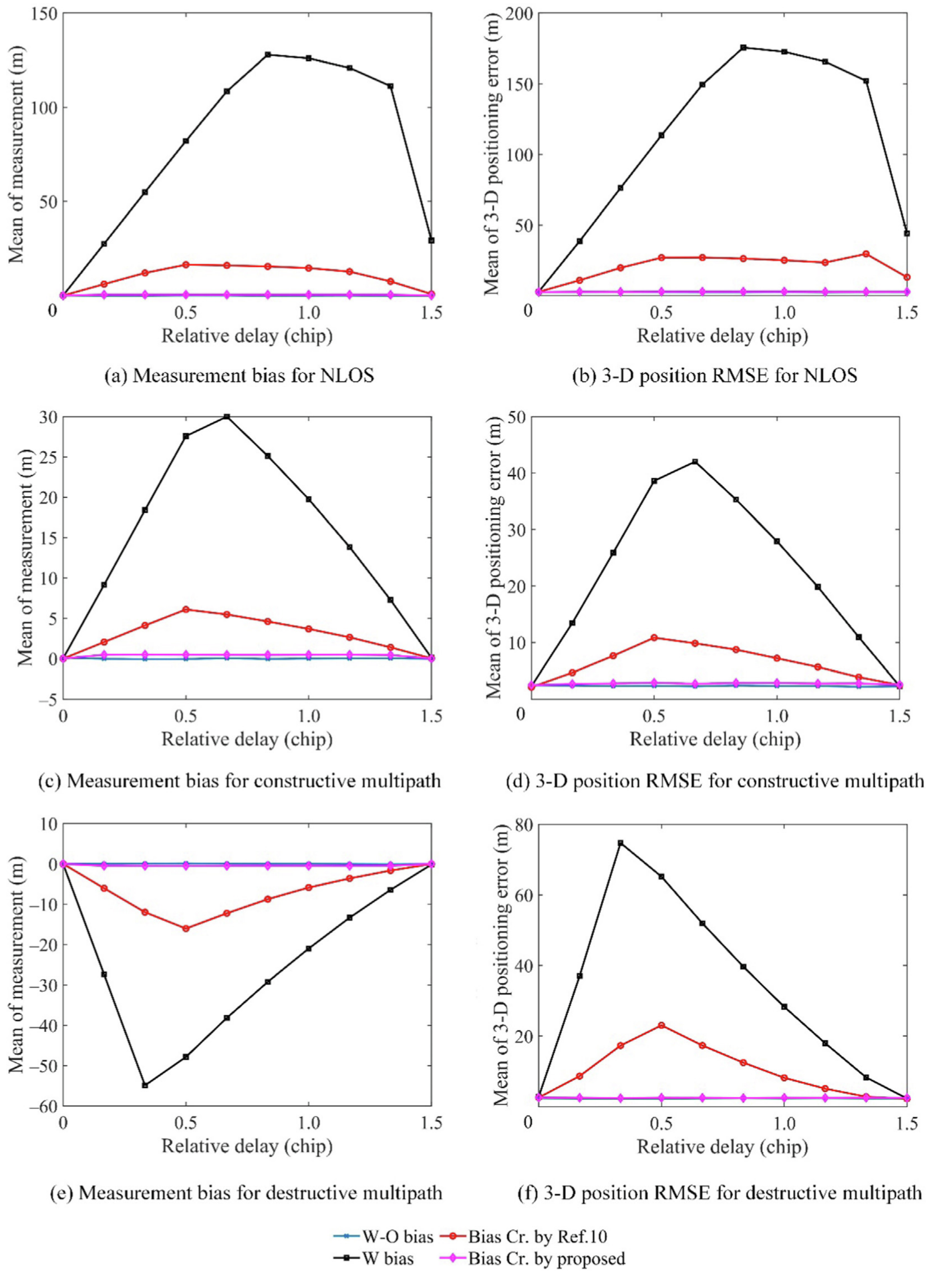
(d) Bias corrected by proposed method

(e) 3-D positioning error comparison

(f) Clock bias error comparison

Fig. 7 Comparison of the tracking and navigation performance as a NLOS appears during 8–12 s. The curve "W-O bias" is the result from a normal signal reception without multipath/NLOS; hence, the measurement is zero mean. The curve "W bias" stands for the result from one channel with multipath/NLOS reception, but without bias correction; the curve "Bias Cr. By Ref. 10" denotes the result from the channel with multipath/NLOS reception, and with the tracking bias corrected by the method in Ref. 10; the curve "Bias Cr. by proposed" denotes the tracking bias is corrected by the proposed method.

tracking bias is defined as the summation of the absolute mean of the tracking bias from the channels except the NLOS reception channel. The total propagated tracking bias in Fig. 9(a) is 0.57 chip, while the ones after correction of tracking bias with the method in Ref. 10 and Fig. 9(b) and the proposed method in Fig. 9(c) are 0.087 chip and 0.062 chip, respectively. The mean of the code discriminator output from the NLOS channel without bias correction in Fig. 9(a), after tracking bias corrected by the method in Ref. 10 and Fig. 9(b) and the proposed method in Fig. 9(c) are 0.406 chip, 0.176 chip and 0.167 chip, respectively. The horizontal positioning results on the bottom panel also shows the superiority of the proposed bias correction method, where the horizontal positioning error in the three cases are 86.798 m, 22.294 m, and 16.732 m.



(a) Measurement bias for NLOS

(b) 3-D position RMSE for NLOS

(c) Measurement bias for constructive multipath

(d) 3-D position RMSE for constructive multipath

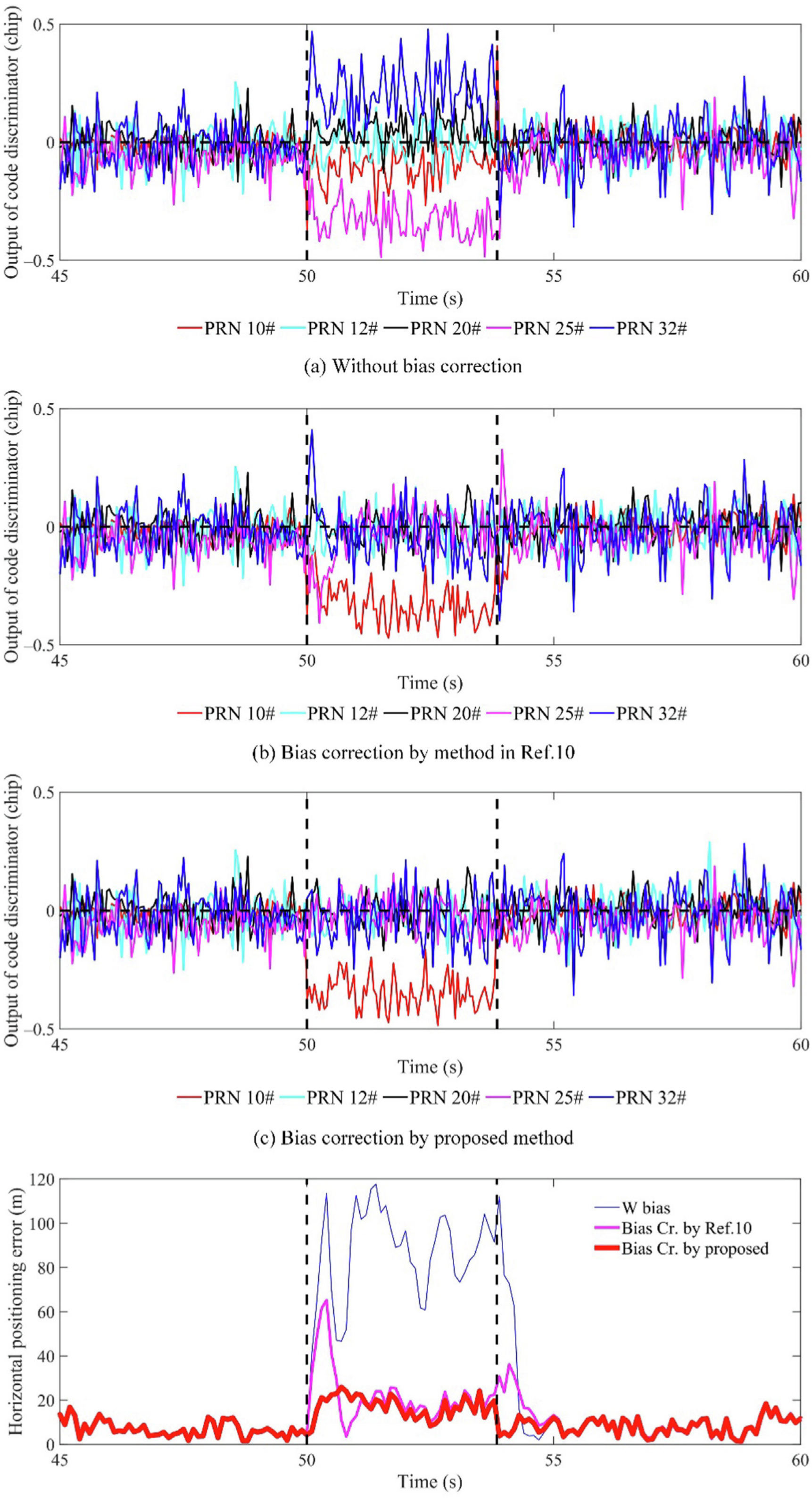
(e) Measurement bias for destructive multipath

(f) 3-D position RMSE for destructive multipath

Fig. 8 Performance comparison as multipath appears.

6. Conclusions

Conclusions are drawn by analyzing four kinds of potential biases in the two existing VTL implementations. Both of the initial user state bias and the initial code phase bias can be eliminated by the new measurements in the absolute VTL, but they cannot be eliminated in the relative VTL. The user state bias induced from an erroneous navigation solution during VTLs' operational period can be eliminated by both of the two VTL implementations. The appearance of the multipath/NLOS introduces tracking bias and then the bias propagates, while the induced bias disappears as the signal recovers. The case is true for both of the VTL implementations. In the duration of the multipath/NLOS, bias propagation would decrease the bias correction performance. Hence, it is important to correct the tracking bias before its propagation to ensure the per-



(a) Without bias correction

(b) Bias correction by method in Ref.10

(c) Bias correction by proposed method

(d) Comparison of horizontal positioning error

Fig. 9 Performance comparison of different methods for a field collected data with a manually added NLOS reception.

formance of VTLs during the multipath/NLOS period. Based on the above conclusions, a robust VTL structure is proposed, where the absolute VTL is selected for its robustness to the two initialization biases; meanwhile, the instant bias detection and correction method is used to eliminate the multipath/NLOS induced bias in its duration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study was co-supported by the Scientific Research Program of Tianjin Municipal Education Commission, China (No. 2021KJ042), and the Special Project of the National Science Foundation of China (No. U2133204).

References

1. Kaplan ED, Hegarty C. *Understanding GPS principles and applications*. 2nd ed. London: Artec House; 2017.
2. Teunissen PJG, Montenbruck O. *Springer handbook of global navigation satellite systems*. 1st ed. Cham: Springer International Publishing; 2017.
3. Kuusniemi H, Wieser A, Lachapelle G, et al. User-level reliability monitoring in urban personal satellite-navigation. *IEEE Trans Aerosp Electron Syst* 2007;**43**(4):1305–18.
4. Kuusniemi H, Lachapelle G, Takala JH. Position and velocity reliability testing in degraded GPS signal environments. *GPS Solut* 2004;**8**(4):226–37.
5. Jia QQ, Wu RB, Wang WY, et al. Multipath interference mitigation in GNSS via WRELAX. *GPS Solut* 2016;**21**(2):487–98.
6. Lashley M, Bevly DM. *Analysis of discriminator based vector tracking algorithms*. San Diego: Springer; 2007. p. 570–6.
7. Lashley M, Bevly DM, Hung JY. Performance analysis of vector tracking algorithms for weak GPS signals in high dynamics. *IEEE J Sel Top Signal Process* 2009;**3**(4):661–73.
8. Lashley M, Bevly DM, Hung JY. A valid comparison of vector and scalar tracking loops. *IEEE/ION position, location and navigation symposium*. Piscataway: IEEE Press; 2010. p. 464-74.
9. Jia QQ, Hsu LT, Wu RB. Scalar and vector tracking loop simulation based on a uniform semi-analytic model and robustness analysis in multipath/NLOS situations. *GPS Solut* 2022;**26**(3):1–12.
10. Xu B, Jia QQ, Hsu LT. Vector tracking loop-based GNSS NLOS detection and correction: Algorithm design and performance analysis. *IEEE Trans Instrum Meas* 2020;**69**(7):4604–19.
11. Luo ZB, Zhao L, Ding JC, et al. Tracking error analysis and performance evaluation method for GNSS non-coherent vector tracking loop. *China satellite navigation conference*. Singapore: Springer; 2019. p. 450-62.
12. Kim KH, Song JH, Jee GI. The GPS vector tracking loop based on the iterated unscented unscented Kalman filter under the large initial error. *2009 European control conference (ECC)*. 2009 Aug 23-26. Piscataway: IEEE Press; 2009. p. 3701-6.
13. Xu B, Hsu LT. Open-source MATLAB code for GPS vector tracking on a software-defined receiver. *GPS Solut* 2019;**23**(2):1–9.
14. Zhao SH, Lu MQ, Feng ZM. Implementation and performance assessment of a vector tracking method based on a software GPS receiver. *J Navigation* 2011;**64**(S1):S151–61.
15. Zou XJ, Lian BW, Wu P, et al. Detect and remove the blocked channel in the vector tracking loop based on carrier to noise density ratio. *Proceedings of the 30th international technical meeting of the satellite division of the institute of navigation (ION GNSS+ 2017)*. 2017. p. 3648-60.
16. Bhattacharyya S, Gebre-Egziabher D. Vector loop RAIM in nominal and GNSS-stressed environments. *IEEE Trans Aerosp Electron Syst* 2014;**50**(2):1249–68.
17. Bhattacharyya S, Gebre-Egziabher D. Integrity monitoring with vector GNSS receivers. *IEEE Trans Aerosp Electron Syst* 2014;**50**(4):2779–93.
18. Bhattacharyya S, Mute DL, Gebre-egziabher D. Kalman filter-based reliable GNSS positioning for aircraft navigation. *Proceedings of the AIAA* scitech *2019 forum*. Reston: AIAA; 2019.
19. Bhattacharyya S, Gebre-Egziabher D. Development and validation of parametric models for vector tracking loops. *J Inst Navigation* 2010;**57**(4):275–95.
20. Bhattacharyya S. Vector loop transfer functions and noise bandwidths. *J Inst Navigation* 2018;**65**(1):55–72.
21. Sun ZY, Wang XL, Feng SJ, et al. Design of an adaptive GPS vector tracking loop with the detection and isolation of contaminated channels. *GPS Solut* 2016;**21**(2):701–13.
22. Hsu LT, Tokura H, Kubo N, et al. Multiple faulty GNSS measurement exclusion based on consistency check in urban canyons. *IEEE Sens J* 2017;**17**(6):1909–17.
23. Lin HL, Huang YB, Tang XM, et al. A robust vector tracking loop based on diagonal weighting matrix for navigation signal. *Adv Space Res* 2017;**60**(12):2607–19.
24. Kim KH, Jee GI, Im SH. Adaptive vector-tracking loop for low-quality GPS signals. *Int J Contr Autom Syst* 2011;**9**(4):709–15.
25. Amani E. Scalar and vector tracking algorithms with fault detection and exclusion for GNSS receivers design and performance evaluation [dissertation]. Paris: Université Paris-Est; 2017 [French].
26. Amani E, Djouani K, De Boer JR, et al. Adaptive and conjoint scalar-vector tracking loops for GNSS tracking robustness and positioning integrity. *2017 European navigation conference (ENC)*; 2017 May 9-12; Lausanne, Switzerland. Piscataway: IEEE Press; 2017.
27. Jwo DJ, Wen ZM, Lee YC. Vector tracking loop assisted by the neural network for GPS signal blockage. *Appl Math Model* 2015;**39**(19):5949–68.
28. Jwo DJ, Wen ZM. Neural network assisted vector tracking loop for bridging GPS signal outages. *Appl Mech Mater* 2015;**764–765**:560–4.
29. Zhao JM, Zhao XF, Li DA, et al. GPS/BDS VTL-assisted by the NN for complex environments. *IET Commun* 2018;**12**(4):473–9.
30. Hsu LT, Jan SS, Groves PD, et al. Multipath mitigation and NLOS detection using vector tracking in urban environments. *GPS Solut* 2014;**19**(2):249–62.
31. Jiang CH, Xu B, Hsu LT. Probabilistic approach to detect and correct GNSS NLOS signals using an augmented state vector in the extended Kalman filter. *GPS Solut* 2021;**25**(2):1–14.
32. Jiang CH, Chen YW, Xu B, et al. Vector tracking based on factor graph optimization for GNSS NLOS bias estimation and correction. *IEEE Internet Things J* 2022;**9**(17):16209–21.
33. Lesouple J, Robert T, Sahmoudi M, et al. Multipath mitigation for GNSS positioning in an urban environment using sparse estimation. *IEEE Trans Intell Transp Syst* 2018;**20**(4):1316–28.
34. Jwo DJ, Wang SH. Adaptive fuzzy strong tracking extended Kalman filtering for GPS navigation. *IEEE Sens J* 2007;**7**(5):778–89.