ORIGINAL ARTICLE

Random forest-based multipath parameter estimation

Xin Qi[1] · Bing Xu[1] · Zhipeng Wang[2] · Li-Ta Hsu[1]

Abstract

Multipath is recognized as one of the major error sources for GNSS urban navigation. This study proposes a random forest (RF)-based multipath parameter estimator that uses random forest regression for parameter estimation, thereby mitigating multipath effect by removing the estimated reflected signal components. The proposed estimator is evaluated and compared with the multipath estimation delay-lock loop (MEDLL) for one-multipath and three-multipath cases, respectively. Simulation results demonstrate that the RF-based estimator is less affected by the front-end bandwidth of received signals, compared with MEDLL. The proposed RF-based estimator shows better performance than MEDLL for signals with front-end bandwidths of lower than 6 MHz. In 20 sets of tests on signals with a front-end bandwidth of 10 MHz in the three-multipath case, the RF-based estimator obtains smaller standard deviations than MEDLL. In experiments using real data with a front-end bandwidth of 2 MHz, the RF-based estimator reduces the 2D and 3D positioning errors by 8.5% and 8.7% over 180 epochs, respectively, against the conventional delayed-locked loop (DLL).

Keywords Random forest (RF) · Global Navigation Satellite System (GNSS) · Multipath mitigation · Multipath estimation delay-lock loop (MEDLL)

Introduction

Global Navigation Satellite System (GNSS) is widely used in various fields, such as precision agriculture (Du et al. 2023), smart transportation (Anand et al. 2021) and autonomous vehicles (de Miguel et al. 2020). Multipath has long been recognized as one of the major error sources for urban GNSS navigation and positioning as it is highly related to the environments. When multipath interference occurs, reflected signals superimposed on the direct-path (DP) signal can cause distortion to the autocorrelation function, and hence add a bias to the pseudorange measurement. In order to mitigate the effect of multipath, many measures have been proposed such as advanced antenna techniques (Xie et al. 2017; Granger et al. 2008; Wang et al. 2019) and baseband signal processing techniques (Van Dierendonck et al. 1992; Garin 1996; McGraw et al. 1999; Lee 2002; Wu et al. 2012;

Van Nee et al. 1994; Weill 2002). Three-dimensional (3D) building models are also increasingly used to aid navigation and improve localization accuracy in densely built-up areas as an effective means of multipath and no-line-of-sight (NLOS) mitigation (Zhang et al. 2020).

In recent years, a large number of multipath mitigation methods at baseband signal processing level have been proposed. These methods can be broadly grouped into two categories: (1) advanced correlator and discriminator design that directly obtains the result after the multipaths are suppressed or eliminated, such as narrow correlator (Van Dierendonck et al. 1992), multipath elimination technology (MET) (Townsend et al. 1994), strobe edge correlator (Garin 1996), high resolution correlator (HRC) (McGraw et al. 1999) and code correlation reference waveforms (CCRW) technology (Lee 2002; Wu et al. 2012), etc.; and (2) multipath parameter estimation which estimates individual multipath components and subtracts them from the total received signal to restore the DP signal, such as the multipath estimating delay-lock loop (MEDLL) (Van Nee et al. 1994), the vision correlator (Fenton et al. 2005) and the multipath mitigation technique (MMT) (Weill 2002).

MEDLL was proposed by NovAtel in 1994. It uses multiple correlators to sample a series of correlation values

✉ Bing Xu
  pbing.xu@polyu.edu.hk

1 Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, China

2 School of Electronics and Information Engineering, Beihang University, 37 Xueyuan Road, Beijing, China

and obtain the correlation peak envelop. MEDLL estimates multipath signal parameters based on maximum likelihood estimation criterion. As an effective multipath mitigation method, MEDLL can eliminate about 90% multipath effect in the narrow correlation receiver in some cases (Xie 2009). However, the maximum likelihood estimation criterion determines that the performance of MEDLL depends heavily on the shape of the autocorrelation function and is therefore undoubtedly affected by the front-end bandwidth setting and the carrier-to-noise ratio (CNR) of the received signal.

Machine learning has been widely used in GNSS field, such as spoofing detection (Semanjski et al. 2019; Aissou et al. 2022), signal acquisition (Borhani-Darian et al. 2023) and positioning enhancement (Kanhere al. 2022), etc. In recent years, machine learning has also been utilized to solve multipath interference. On one hand, various intelligent methods for multipath detection have been proposed. Hsu (2017) proposed a support vector machine (SVM)-based LOS/MP/NLOS classifier which use signal strength and pseudorange residue as the features; Quan et al. (2018) employed conventional neural network (CNN) for feature extraction in multipath detection for kinematic GPS positioning; Savas et al. (2019) proposed a multipath detection algorithm based on K-means clustering so that no priori training data is required; Suzuki et al. (2020) constructed a machine learning-based multipath/NLOS classifier that discriminates NLOS multipath signals from the output of the multiple GNSS signal correlators of a software GNSS receiver.

At a deeper level, scholars have attempted to employ machine learning algorithms to mitigate the effect of multipath directly. Phan et al. (2013) employed support vector regression (SVR) to estimate multipath error by making use of the connections between multipath error and satellite relative elevation angle and azimuth angle. In baseband signal processing level, Orabi et al. (2020) proposed a neural network-based delay-locked loop (NNDLL), which built a neural network discriminator using multilayer perceptron (MLP) to estimate code phase directly. Li et al. (2022) proposed a deep network correlator, where a convolutional layer is used to achieve standard correlation, and then a MLP is employed to filter out the multipath effects on the autocorrelation function. These two attempts show the potential of machine learning assistance at the baseband signal processing level. However, both of them focus on correlators or discriminators design, while the application of machine learning to multipath parameter estimation is less investigated. Utilizing the powerful learning capability of machine learning, this study aims to propose a machine learning-based multipath parameter estimator to achieve better robustness than traditional multipath parameter estimation algorithms such as MEDLL. Besides, both the NNDLL and the deep network correlator employ a large number of correlators, which limits the feasibility of these

algorithms in receivers. Therefore, the balance between algorithm performance and computational effort is also an issue to be considered.

We develop a random forest (RF)-based multipath parameters estimator, where random forest regression is employed to estimate the parameters of multipath for GNSS signals. In (Qi et al. 2023), we have presented the preliminary results of parameter estimation and multipath mitigation performance evaluation of the RF-based estimator in the one-multipath case. We present tests of the algorithm in greater depth and detail in this paper. First, the multipath mitigation performance of the RF-based estimator is evaluated using MEDLL as a benchmark in one-multipath and three-multipath simulation cases, respectively. The effect of both CNR and front-end bandwidth setting of received signals are considered simultaneously. The simulation results show that both MEDLL and the RF-based estimator can mitigate the effect of multipath to some extent compared to no multipath correction case. Compared to MEDLL, the RF-based estimator is rarely affected by the front-end bandwidth and shows better robustness. In the experiments with real data, the delay-locked loop (DLL) with RF-based estimator achieves both lower root mean square error (RMSE) and lower standard deviation of positioning errors than conventional DLL and MEDLL, which again validates the effectiveness of our algorithm. We also test the performance of the RF-based parameter estimator considering different numbers of correlators. The results show that the proposed algorithm works effectively even when only 21 correlators are used.

The following sections are organized as follows. First, the GNSS signal model in the presence of multipath we considered is introduced. Then, we present the methodology and the architecture of the RF-based multipath parameter estimator and how it works in a DLL. After that, simulation results comparing the performance of the RF-based estimator and MEDLL in one-multipath and three-multipath cases are presented and discussed. In addition, the multipath mitigation performance of the DLL with RF-based estimator is investigated with real data. Finally, the conclusion is illustrated.

Signal model

The down-converted intermediate-frequency signal at the receiver can be modeled as

$$r_\Sigma(t) = \sum_{i=0}^{M} a_i p\left(t - \tau_i\right) \cos\left(2\pi f_{IF} t + \phi_i\right) + n(t) \qquad (1)$$

where the 0-th signal is the DP signal; $p(t)$ is the pseudo-random noise (PRN) code of a specific satellite; a_i, τ_i and ϕ_i are the amplitude, code phase and carrier phase of the i-th signal; f_{IF} is the intermediate frequency. The amplitude of DP signal a_0 is assumed to be unity. The relative amplitude

of i-th reflected signal $a_i \in (0, 1)$ considering the attenuation because of reflections. $n(t)$ is the measurement noise process, which can be modeled as additive white Gaussian noise (AWGN).

Correlating the received signal with local replica yields

$$R_\Sigma(t) = \sum_{i=0}^{M} a_i R(t - \tau_i) e^{j\phi_i} + N(t) \tag{2}$$

where $R(t - \tau_i)$ is the autocorrelation function of the PRN code shifted by code phase delay τ_i; $N(t)$ is the low-pass filtered noise.

$R_\Sigma(t)$ is a complex signal whose real and imag parts are realized in two arms in a receiver. Since the specific value of carrier phase is not concerned in the code tracking loop, only the real part of $R_\Sigma(t)$ is employed, shown as

$$R_{\Sigma I}(t) = R(t - \tau_0) + \sum_{i=1}^{M} a_i R(t - \tau_i) \cos \overline{\phi}_i + N_I(t) \tag{3}$$

where $\overline{\phi}_i$ is the relative carrier phase of i-th reflected signal with respect to the DP signal.

For each signal component, the change in carrier phase acts on the amplitude of the autocorrelation function. Here we perform an invertible transformation, letting

$$\alpha_{Ii} = a_i \cos \overline{\phi}_i \tag{4}$$

Then, equation (3) is transferred to

$$R_{\Sigma I}(t) = R(t - \tau_0) + \sum_{i=1}^{M} \alpha_{Ii} R(t - \tau_i) + N_I(t) \tag{5}$$

For the $R_{\Sigma I}(t)$ shown in (5), the multipath parameters to be estimated at time k are $\theta^k = \{\alpha_{I1}, \cdots, \alpha_{IM}, \tau_1, \cdots, \tau_M\}$. Without considering noise, the autocorrelation function of the DP signal can be restored by

$$R(t - \tau_0) = R_{\Sigma I}(t) - \sum_{i=1}^{M} \hat{\alpha}_{Ii} R(t - \hat{\tau}_i) \tag{6}$$

where $\hat{\alpha}_{Ii}, \hat{\tau}_i \in \hat{\theta}^k$ is the parameter estimates of i-th multipath at time k.

Methodology

Multipath parameter estimation is essentially a regression problem. We employ random forest regression to construct the multipath parameter estimator. This section describes the principle of random forest regression and how the RF-based parameter estimator works in the code tracking loop.

Random forest regression

Random forest is an ensemble learning algorithm proposed by Breiman in 2001 (Breiman 2001). As a successful general-purpose classification and regression method, it can be applied to a wide range of prediction problems and has just a few parameters to tune (Biau et al. 2016). RF consists of multiple decision trees as base learners that operate as an ensemble. A technique called bagging (a contraction of bootstrap-aggregating) (Breiman 1996) is implemented in RF. Bagging creates multiple subsets of the original data by sampling with replacement, and trains a base learner on each subset. The final decision of RF would be obtained by aggregating the predictions of all the base learners. For classification tasks, the output of RF is the class selected by most trees. In the case of a regression problem, the final output is obtained by averaging of all the predictions from individual decision trees.

The general framework of random forest regression is nonparametric regression estimation whose input is a random vector $X \in \chi \subset \mathbb{R}^n$, and the goal is to predict the response $Y \in \mathbb{R}$ by estimating the regression function $m(x) = \mathbb{E}\{Y|X = x\}$. In regression, mean square error (MSE) shown in (7) is often chosen as the impurity function (IF), which is used to determine the split nodes of decision trees.

$$IF = \frac{1}{n_j} \sum_{i=1}^{n_j} \left(Y_i - \hat{Y}_i \right)^2 \tag{7}$$

where n_j is the number of samples in the input space.

For a training dataset $D = \{ (X_1, Y_1), \cdots, (X_n, Y_n) \}$ of random variables distributed as the independent prototype pair (X, Y), the goal of a random forest regression is to construct an estimate $m_D : \chi \to Y$, which satisfy

$$\lim_{n \to \infty} \mathbb{E} \left[m_D(X) - m(X) \right]^2 \to 0 \tag{8}$$

The training dataset of j-th decision tree D_j is obtained by sampling with replacement from the original dataset. The classification and regression trees (CART)-split criterion (Breiman et al. 1984) is used to construct individual trees by choosing the best split nodes. Suppose s is the split node for a feature variable q, the input space is split into two subspace $R_1(q,s) = \{X|X^{(q)} \leq s\}$ and $R_2(q,s) = \{X|X^{(q)} > s\}$. The goal of CART regression is to find the optimal (q^*, s^*), which minimizes the sum of $IF(R_1)$ and $IF(R_2)$. The optimization function in CART regression is written as

$$J(q,s) = \min_{c_1} \sum_{X_i \in R_1(q,s)} \left(Y_i - c_1 \right)^2 + \min_{c_2} \sum_{X_i \in R_2(q,s)} \left(Y_i - c_2 \right)^2 \tag{9}$$
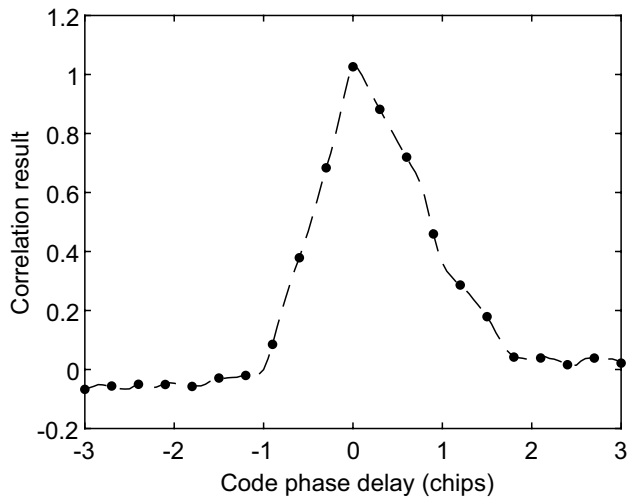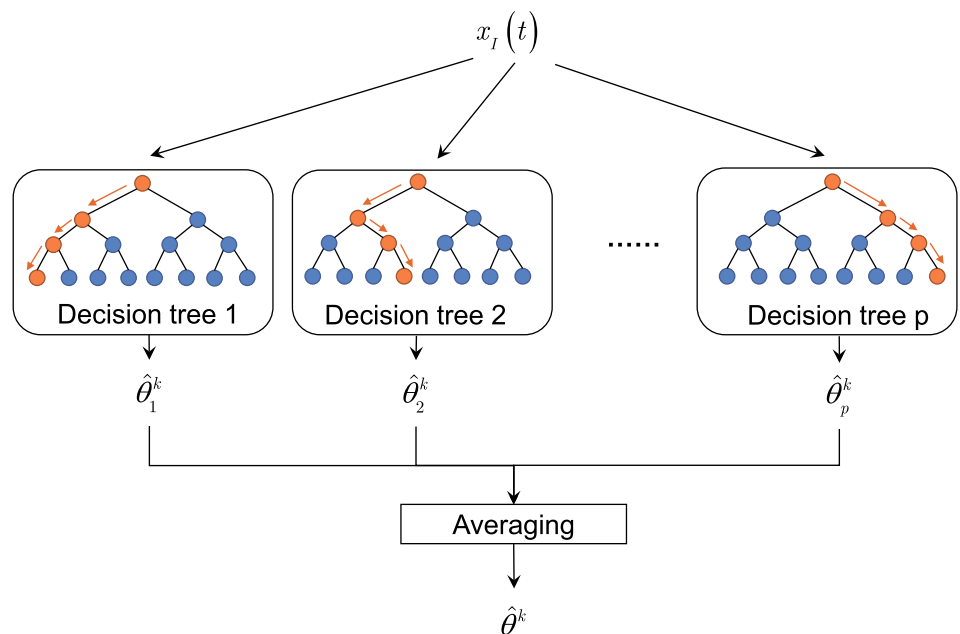
where

Fig. 1 Autocorrelation function with one constructive multipath

$$\begin{cases} \hat{c}_1 = ave\big(Y_i | X_i \in R_1(q,s)\big) \\ \hat{c}_2 = ave\big(Y_i | X_i \in R_2(q,s)\big) \end{cases} \tag{10}$$

The above partitioning process would be repeated for each subspace until the stopping condition is satisfied, then a decision tree is generated. Assuming that the whole input space is finally divided into M regions $R_1, R_2, \cdots, R_M$. After training, the prediction of the j-th decision tree $m_{D_j}(x)$ with x as the input is

$$m_{D_j}(x) = \sum_{m=1}^{M} \hat{c}_m I\big(x \in R_m\big) \tag{11}$$

where

$$\hat{c}_m = ave\big(Y_i | x_i \in R_m\big) \tag{12}$$

The output of the random forest regression is calculated by

$$m_D(x) = \frac{1}{p} \sum_{j=1}^{p} m_{D_j}(x) \tag{13}$$

where p is the number of decision trees in the random forest.

Random forest-based multipath parameter estimator

When the multipath interference occurs, there is distortion in the autocorrelation function. The dashed line in Fig. 1 depicts an autocorrelation function for constructive multipath interference. On the other hand, the distorted autocorrelation function contains all the information of the multipath, such as relative amplitude and code delay with respect to the DP signal. Therefore, the equally spaced sampling points of the autocorrelation function are chosen as the input features $x_I(t)$ of the random forest for multipath parameter estimation. The points in Fig. 1 illustrate the sampling points used as features. The architecture of the RF-based parameter estimator is shown in Fig. 2. $\hat{\theta}$ at time k is calculated by

$$\hat{\theta}^k = \frac{1}{p} \sum_{j=1}^{p} \hat{\theta}_j^k \tag{14}$$

where $\hat{\theta}_j^k$ is the prediction from the j-th decision tree at time k.

Fig. 2 Diagram of RF-based parameter estimator. p is the number of decision trees

The architecture of tracking loops with the RF-based parameter estimator is shown in Fig. 3. The equally spaced sampling points of the autocorrelation function are implemented by a multi-correlator. The multi-correlator outputs are then sent to the RF-based estimator to calculate multipath parameter estimates $\hat{\theta}^k$. After that, the autocorrelation function components of reflected signals are restored by $\hat{\theta}^k$ and then subtracted from the total autocorrelation function. Note that a fixed reference autocorrelation function R_{ref} is employed to restore the autocorrelation function components. Theoretically, the amplitude of the autocorrelation function of DP signal is the same as the amplitude of R_{ref}, which is unity.

Simulations

In this section, we evaluate the performance of the RF-based multipath parameter estimator through simulations. The classic multipath parameter estimator, MEDLL, is used as a benchmark. MEDLL uses maximum likelihood estimation criterion to estimate multipath parameter with multiple sampling of complex autocorrelation functions. It estimates the relative amplitude, relative carrier phase and code delay of the multipath signal separately. For the $R_\Sigma(t)$ given in (2), the estimates for i-th signal in MEDLL are shown in (15) (Van Nee et al. 1994)

$$
\begin{cases}
\hat{\tau}_i = \max_\tau \mathfrak{R}\left[\left(R_\Sigma(t) - \sum_{\substack{m=0 \\ m \neq i}}^{M} \hat{a}_m R(t - \hat{\tau}_m) e^{j\hat{\phi}_m}\right) e^{-j\hat{\phi}_i}\right], \\[2em]
\hat{a}_i = R\left[\left(R_\Sigma(\hat{\tau}_i) - \sum_{\substack{m=0 \\ m \neq i}}^{M} \hat{a}_m R(\hat{\tau}_i - \hat{\tau}_m) e^{j\hat{\phi}_m}\right) e^{-j\hat{\phi}_i}\right], \\[2em]
\hat{\phi}_i = \arg\left(R_\Sigma(\hat{\tau}_i) - \sum_{\substack{m=0 \\ m \neq i}}^{M} \hat{a}_m R(\hat{\tau}_i - \hat{\tau}_m) e^{j\hat{\phi}_m}\right).
\end{cases}
\tag{15}
$$

where the notation $\mathfrak{R}(\cdot)$ is the real part of a complex value.

Since the specific value of carrier phase is not calculated separately in our algorithm, only the accuracy of relative amplitude and code phase delay estimates are compared. Note that the relative amplitude estimated by MEDLL is the relative amplitude of the reflected signal with respect to the DP signal a_i; while the relative amplitude calculated by the RF-based estimator is α_{li} defined in (4). Root mean square error (RMSE) is used to assess the accuracy of parameter estimation. The RMSE of τ_i over m moments is shown below:
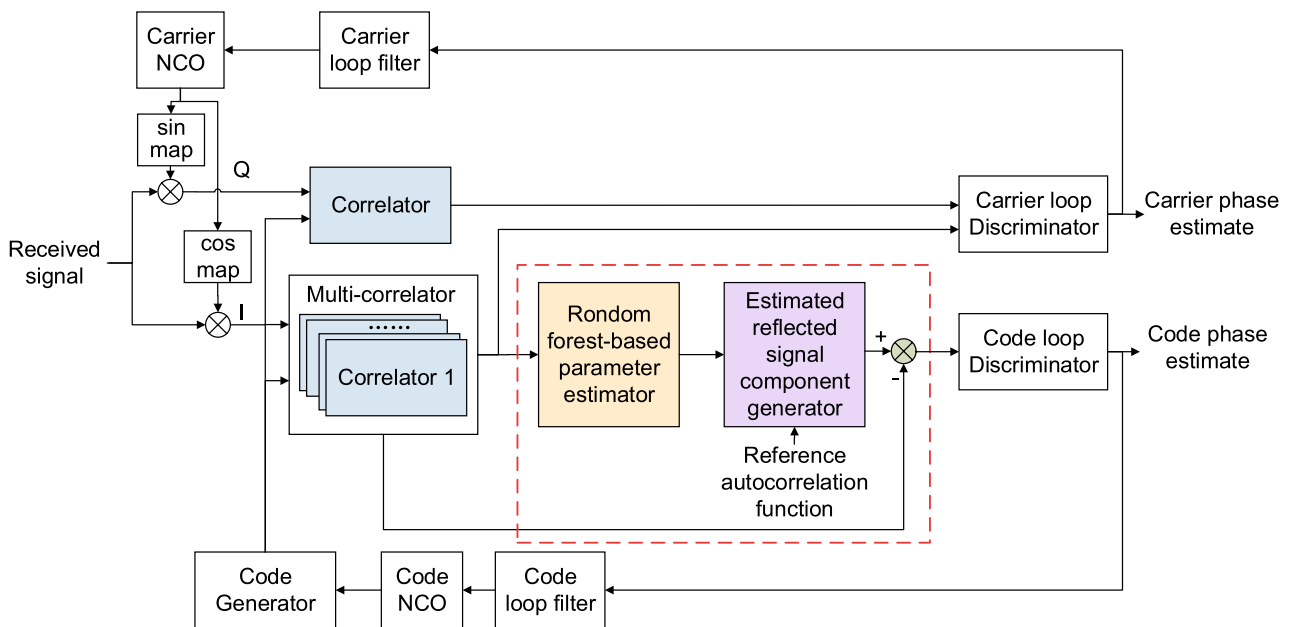

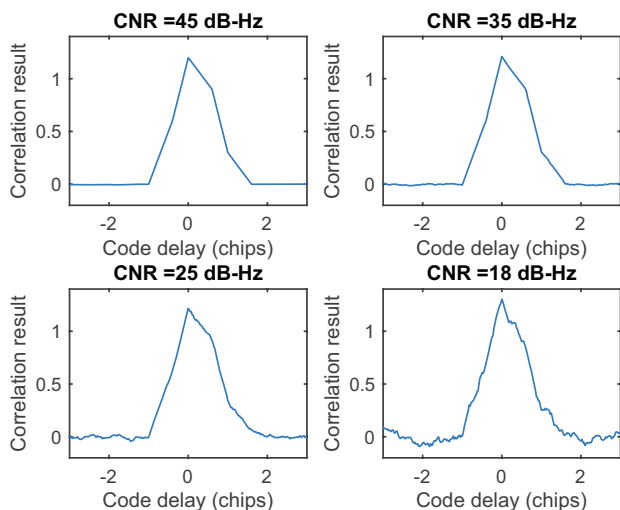**Fig. 3** Architecture of tracking loops with the RF-based multipath parameter estimator

Fig. 4 Autocorrelation functions affected by multipath at different CNRs

Fig. 5 Autocorrelation functions affected by multipath for different front-end bandwidths

$$RMSE(\tau_i) = \sqrt{\frac{1}{m} \sum_{k=1}^{m} \left(\hat{\tau}_i^k - \tau_i^k\right)^2} \tag{16}$$

The shape of the autocorrelation function is influenced by both CNR and front-end bandwidth setting of the received signal. Figures 4 and 5 show the autocorrelation functions subjected to a constructive multipath with different CNRs and different front-end bandwidths. The relative amplitude of the multipath is 0.5 and the relative code delay is 0.6 chips. When the CNR is low, the shape of autocorrelation function will be largely affected by noise. While small front-end bandwidth can filter out high-frequency noise components in the received signal, the high-frequency signal components are also filtered out, causing the rounded autocorrelation function peak. Therefore, both CNR and front-end bandwidth are considered in performance evaluation. A software defined receiver (SDR) developed by Borre et al. (2007) is employed as the test platform.

Performance evaluation in one-multipath case

As a supervised learning algorithm, the performance of random forest regression depends heavily on the quantity and quality of the dataset. We expect the RF-based estimator to be able to accurately estimate the parameters of reflected signals when multipath interference occurs, while not introducing new error sources in multipath-free case. For this purpose, a dataset D_{M1} containing 80,000 samples is generated for the training of random forest, of which 30% is multipath-free samples and 70% is one-multipath samples (including 15% of short multipath samples with code delay less than 0.2 chips and 55% of long multipath samples). The
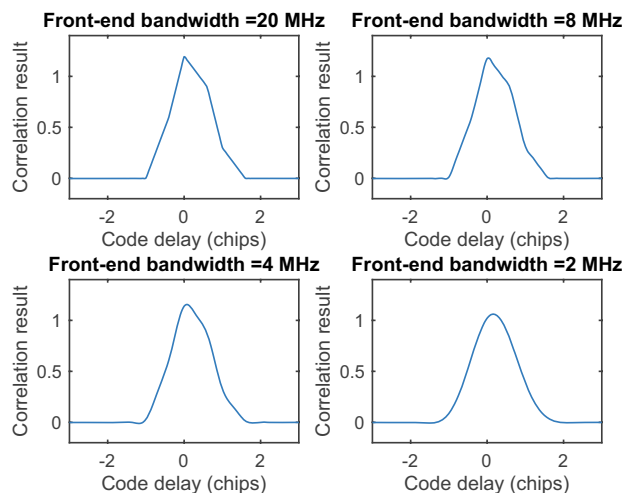
details of multipath parameter settings in D_{M1} are shown in Table 1. The label θ^k of k-th sample in D_{M1} is $\{\alpha_{I1}, \tau_1\}$. The parameters of each sample are sampled uniformly within the range of values. The sample points of the autocorrelation function with 0.3 chips spacing from -3 to $+3$ chip are taken as the features x_I. All these samples in D_{M1} are obtained in simulation for the signal with a sample frequency of 58 MHz and CNRs ranging from 18 to 35 dB-Hz.

The setting of sampling spacing determines the number of correlators required. Considering that it is difficult to implement a large number of correlators in a receiver, the sampling spacing is set to 0.3 chips to balance the feasibility and the performance of the RF-based estimator. There are 21 correlators are required in total in this case. The feasibility of the RF-based estimator in receivers is discussed in detail in discussion.

All 80,000 samples in D_{M1} are used to train the random forest regressor. Each decision tree would be generated by minimizing the optimization function in (9) to find the best split points. Bootstrap aggregating ensures the diversity of base learners. The number of estimators is set to 100, so a random forest regressor consisting of 100 decision trees is generated after training, which is the multipath parameter

Table 1 Details of the multipath parameter settings in one-multipath case training set

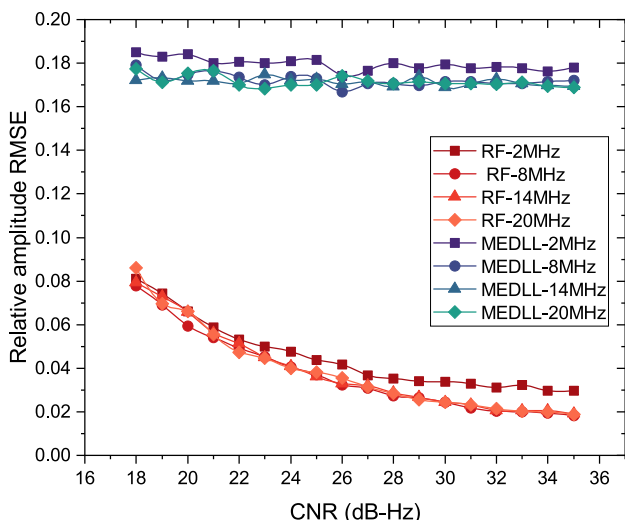| Samples	Percentage	Parameters of the reflected signal		
		a_1	τ_1(chips)	ϕ_1
Multipath-free	30%	/	/	/
One multipath	15%	[0.2, 0.8]	(0, 0.2]	[0, 2π]
	55%	[0.2, 0.8]	(0.2, 2]	[0, 2π]

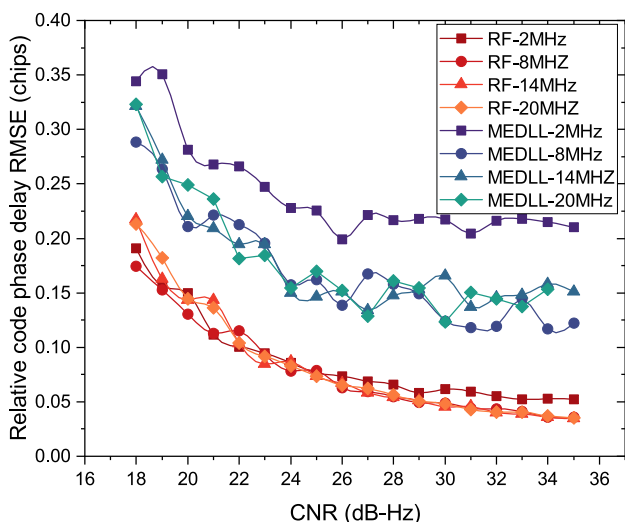**Fig. 6** RMSE of relative amplitude estimates

Fig. 7 RMSE of relative code phase delay estimates

estimator for one multipath case. Details of random forest generation can be found in Methodology.

First, parameter estimation performance of the RF-based estimator is evaluated and compared with MEDLL. 18 test sets are generated with CNRs ranging from 18 to 35 dB-Hz, each consisting of 2000 samples. The multipath parameter settings for the samples in test sets are the same as those in the training set. To explore the impact of front-end bandwidth, 4 series of comparison experiments on parameter estimation are conducted at front-end bandwidth settings of 2 MHz, 4 MHz, 8 MHZ and 20 MHz, respectively. The results are shown in Figs. 6 and 7.

As shown in Figs. 6 and 7, the RF-based parameter estimator achieves both lower amplitude error and lower code phase delay error than MEDLL in all tests. In terms of the

effect of CNR, the estimation error of both relative amplitude and code phase delay obtained by the RF-based estimator increase with the decrease of CNR; the relative code delay estimation errors achieved by MEDLL show the same trend, but the amplitude errors got by MEDLL do not differ much at different CNRs. Concerning the effect of front-end bandwidth, for weak signals with CNR less than 22 dB-Hz, the RF-based estimator obtains smaller parameter estimation errors with low front-end bandwidth settings, while for the signals whose CNR is greater than 22 dB-Hz, the largest parameter estimation errors are obtained for the signal with a front-end bandwidth of 2 MHz. Unlike the results of the RF-based estimator, MEDLL always obtains the largest parameter estimation error for the signals with a front-end bandwidth of 2 MHz, which indicates that the parameter estimation performance of MEDLL performs poorly for signals with low front-end bandwidths.

We then evaluate multipath mitigation performance of the RF-based parameter estimator and compare it with MEDLL. A series of tests for no multipath correction case are performed as the benchmark. All these tests are conducted using signals with different CNRs and front-end bandwidths. At this point, 1 chip spacing normalized early-minus-late (NEML) discriminator, 0.1 chips spacing narrow correlator and 0.05 chips spacing HRC are employed for code discrimination, respectively. The results are shown in Figs. 8, 9, and 10.

As shown in Fig. 8, both MEDLL and the RF-based estimator achieve lower pseudorange error than no multipath correction case when 1 chip spacing NEML discriminator is used for code discrimination. The pseudorange error obtained by both MEDLL and the RF-based estimator increases as the CNR decreases. In terms of the effect of front-end bandwidth, the pseudorange error obtained by MEDLL varies greatly with the front-end bandwidth, whereas the pseudorange error obtained using RF-based estimator is almost stable. Especially, when the front-end bandwidth of received signals is below 6 MHz, the RF-based parameter estimator achieves much lower pseudorange error than MEDLL.

As shown in Fig. 9, the pseudorange error distributions for the cases when the 0.1 chips spacing narrow correlator is used are similar to Fig. 8. However, when the front-end bandwidth is set below 4 MHz, MEDLL produces even larger pseudorange errors than no multipath correction case.

Figure 10 demonstrates the pseudorange error distributions when 0.05 chips spacing HRC is employed for code discrimination. In this test, both MEDLL and the RF-based estimator can still further mitigate the pseudorange error caused by multipath. It is noting that the three subplots in Fig. 10 show similar pseudorange error distributions, and the maximum pseudorange errors due to multipath are at the same level. This is due to the characteristic of HRC.
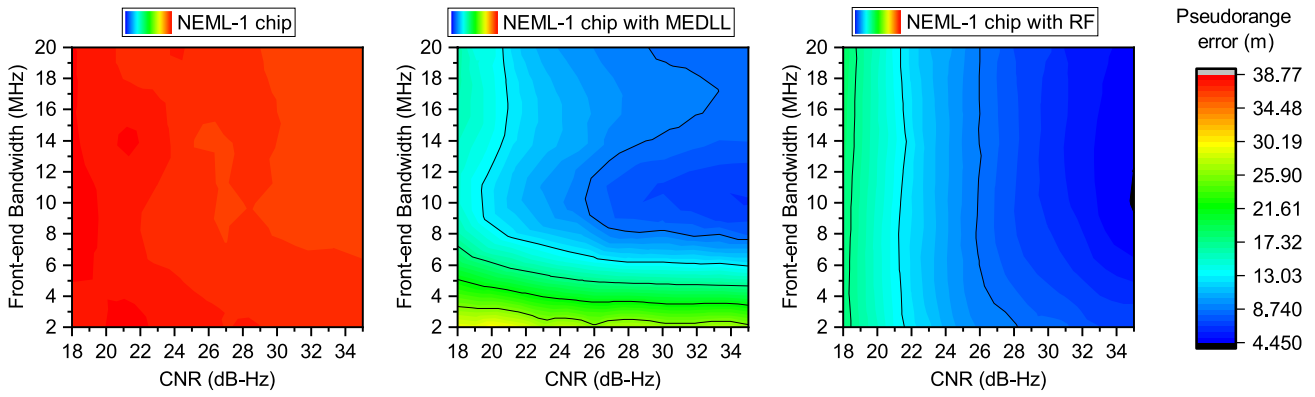
Fig. 8 Color map of the pseudorange error distributions when 1 chip spacing NEML discriminator is used for discrimination in the one-multipath case
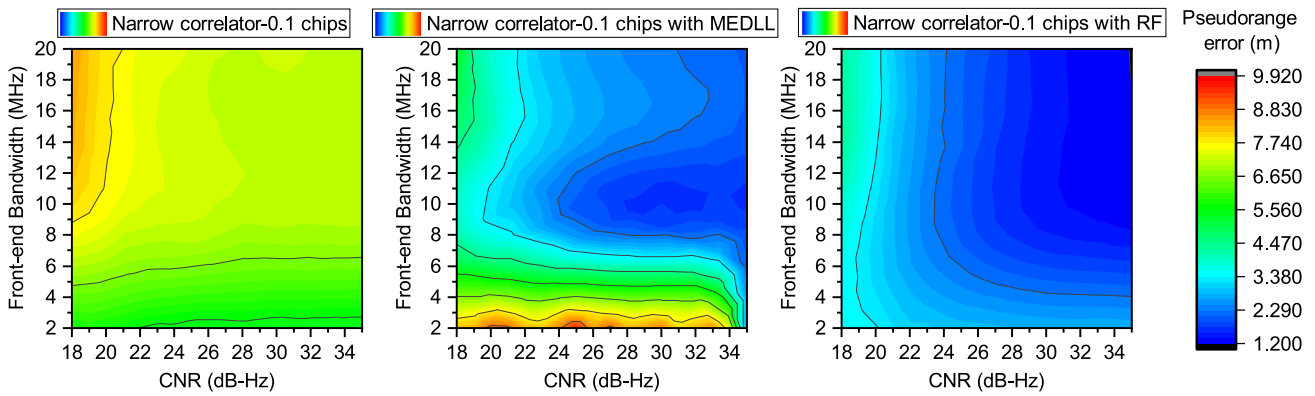

Fig. 9 Color map of the pseudorange error distributions when 0.1 chips spacing narrow correlator is used for discrimination in the one-multipath case
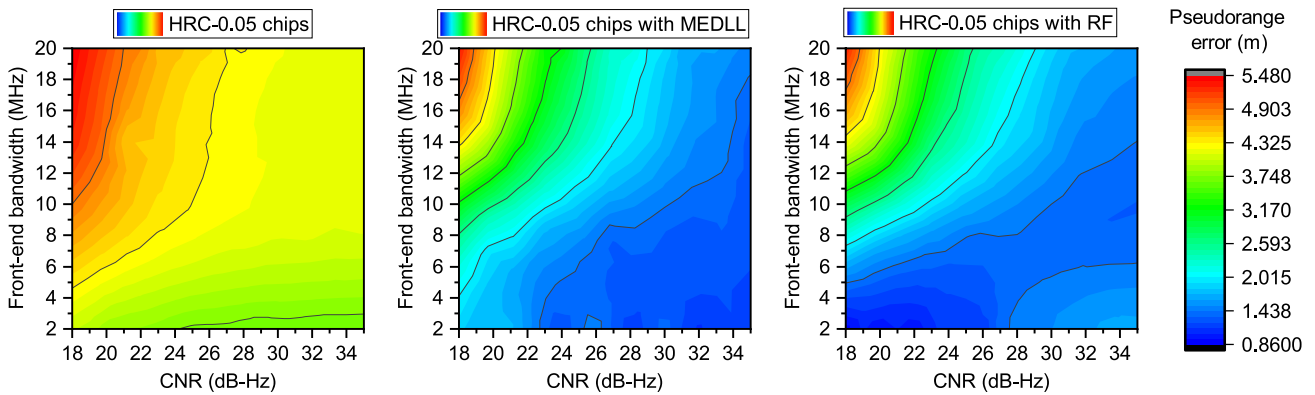

Fig. 10 Color map of the pseudorange error distributions when 0.05 chips spacing HRC is used for discrimination in the one-multipath case

According to (McGraw et al. 1999), HRC is able to eliminate the pseudorange error caused by multipath whose code delay is greater than 0.1 chips.

Both MEDLL and the RF-based estimator can mitigate multipath to some extent, and the multipath mitigation performance of both of them is affected by the CNR of received signal: the larger the CNR, the better the performance. The performance of MEDLL is heavily influenced by the front-end bandwidth setting, whereas RF-based estimator is rarely affected. This can be explained by the fact that

the performance of MEDLL is largely affected by the shape of autocorrelation function. When the front-end bandwidth is low, the noise is filtered out while the high-frequency components in the PRN code are also removed, thus round up the autocorrelation function.

Performance evaluation in three-multipath case

The simulation results in the one-multipath case have verified the superiority of the RF-based multipath parameter estimator. However, there are often multiple reflected signals in the received signal in urban environments. It is essential to test the performance of the proposed algorithm in multiple-multipath cases. Next, the performance of RF-based parameter estimator is evaluated under the three-multipath case and compared with MEDLL. Similarly, a series of simulations without multipath correction are performed as the benchmark.

Considering that the number of reflected signals might be changed due to the relative motion of satellites and the receiver. A dataset D_{M2} containing 80,000 samples is built for RF training. The details of multipath parameter settings in D_{M2} are configured as shown in Table 2. D_{M2} considers multipath-free, one-multipath, two-multipath, and three-multipath examples, which are set up as follows: if the code delay of the first reflected signal is larger than 1.5 chips, the relative amplitude and code delay of the remaining two reflected signals are set to 0; when the delay of the second reflected signal is greater than 1.5 chips, the relative amplitude and code delay of the third reflected signal are set to 0. As with the setup in dataset D_{M1}, the sample points of the autocorrelation function with a spacing of 0.3 chips from -3 to $+3$ chip are taken as inputs x_I, and all these samples are simulated for signals with a sampling frequency of 58 MHz and a CNR of 18 dB-Hz to 35 dB-Hz. The label θ^k of k-th sample in D_{M2} is represented as $\{\alpha_{I1}, \tau_1, \alpha_{I2}, \tau_2, \alpha_{I3}, \tau_3\}$. The multipath parameters of each sample are also sampled uniformly within the range of values. All samples in D_{M2} are used for the training of random forest and the number of estimators is set to 100.

We also consider the effects of CNR and front-end bandwidth. 1 chip spacing NEML discriminator, 0.1 chips spacing narrow correlator and 0.05 chips spacing HRC are used for code discrimination, respectively. The simulation results are shown in Figs. 11, 12 and 13. According to the results, the pseudorange errors in the three-multipath case demonstrate similar distributions as those in the one-multipath case. The difference is that MEDLL obtains best multipath mitigation performance when 1 chip spacing NEML discriminator is used for code discrimination, while the RF-based estimator achieves best performance when using 0.05 chips HRC for discrimination. Besides, it is noted that there are many localized extreme points in the pseudorange distributions realized by MEDLL. To explore why this happens,

Table 2 Details of the multipath parameter settings in the three-multipath case

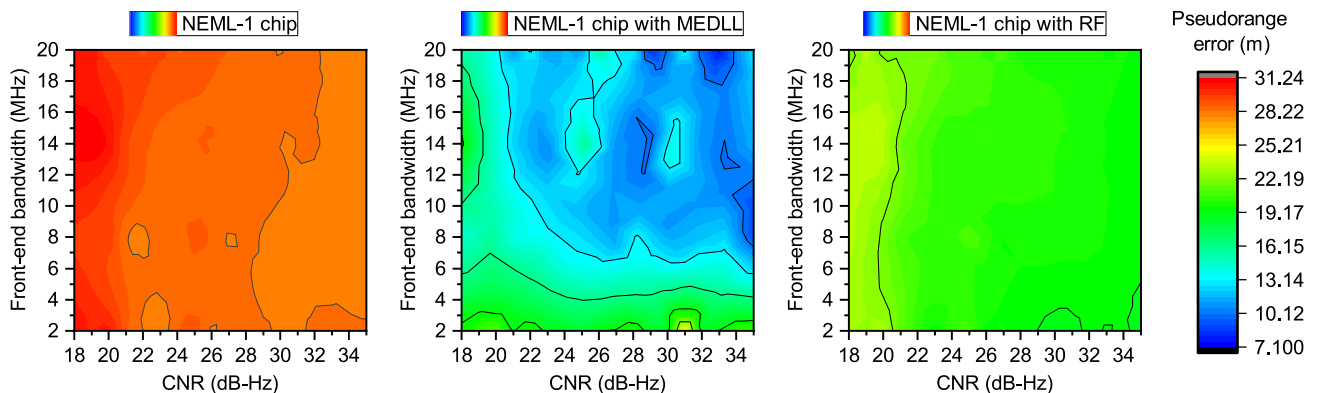| Samples	Percentage	Parameters of the reflected signals			
		Reflected signal no	a_i	τ_i(chips)	ϕ_i
Multipath-free	30%	/	/	/	/
Three multipaths	70%	1	$[0.2, 0.8]$	$(0, 1.8)$	$[0, 2\pi)$
		2	0 or $[0.15, a_1)$	0 or $(\tau_1, 1.95]$	$[0, 2\pi)$
		3	0 or $[0.1, a_2)$	0 or $(\tau_2, 2.0]$	$[0, 2\pi)$

Fig. 11 Color map of the pseudorange error distributions when 1 chip spacing NEML discriminator is used for discrimination in the three-multipath case
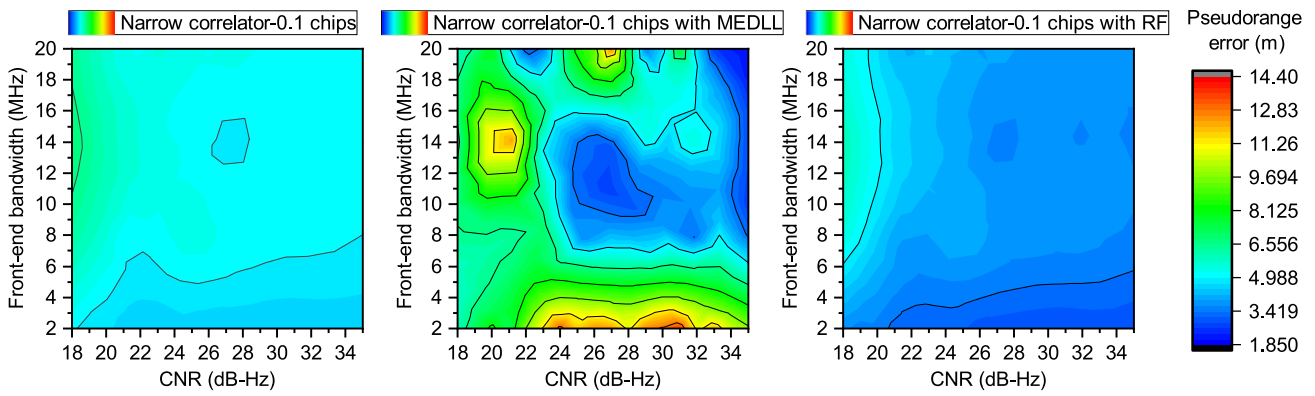
Fig. 12 Color map of the pseudorange error distributions when 0.1 chips spacing narrow correlator is used for discrimination in the three-multipaths case
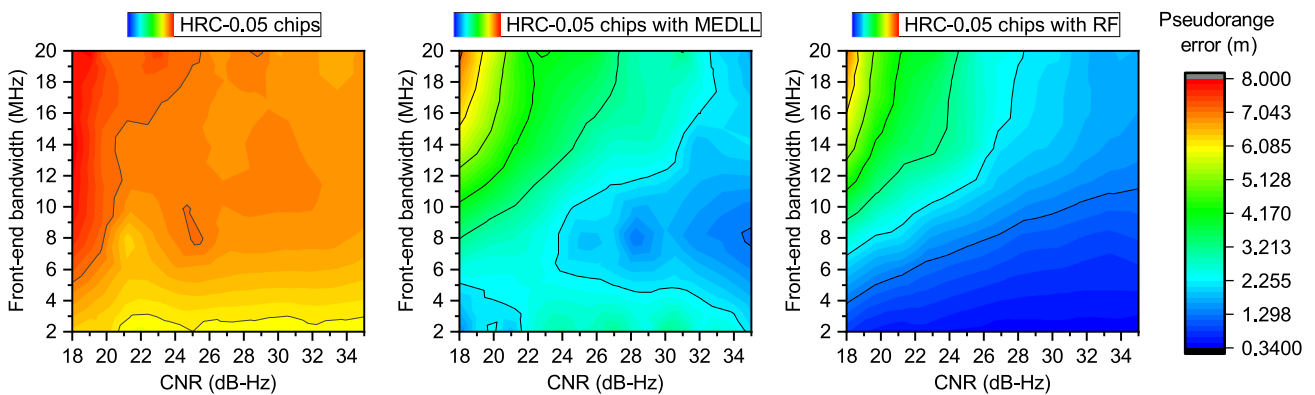

Fig. 13 Color map of the pseudorange error distributions when 0.05 chips spacing HRC is used for discrimination in the three-multipath case

additional tests for a set of signals with a fixed front-end bandwidth are conducted.

Since MEDLL achieves the best performance at a front-end bandwidth of 10 MHz in above simulations. In the following tests, the front-end bandwidth of received signals is set to 10 MHz. Considering the randomness of noise, 20 repetitions of the test are conducted. The results are shown in Figs. 14, 15 and 16. Each point in these figures is the mean value of pseudorange errors in 20 trials, and the error bars are the standard deviations.

According to the results in Figs. 14, 15 and 16, the standard deviations of the pseudorange errors obtained by RF-based estimator and no multipath correction case in all three groups of tests are small, and both the mean value and the standard deviation decrease smoothly with the increase of CNR; whereas the mean value of the pseudorange errors obtained by MEDLL fluctuated largely, and the standard deviations are much larger than those of RF-based estimator and no multipath correction case. These results indicate that the performance of MEDLL is
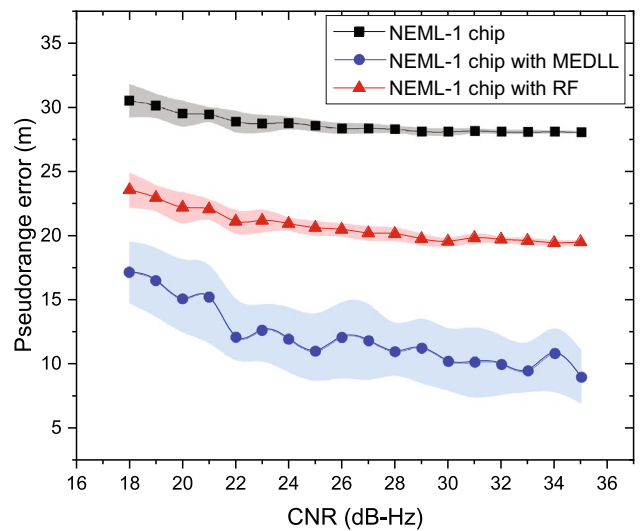

Fig. 14 Pseudorange errors at different CNRs when 1 chip spacing NEML discriminator is used for discrimination
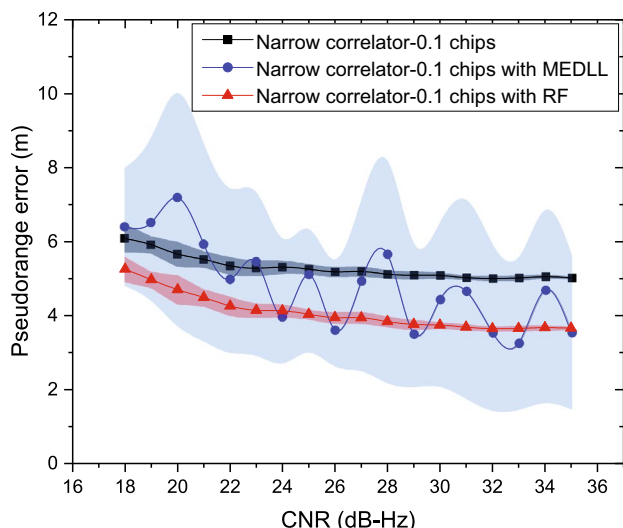
Fig. 15 Pseudorange errors at different CNRs when 0.1 chips spacing narrow correlator is used for discrimination
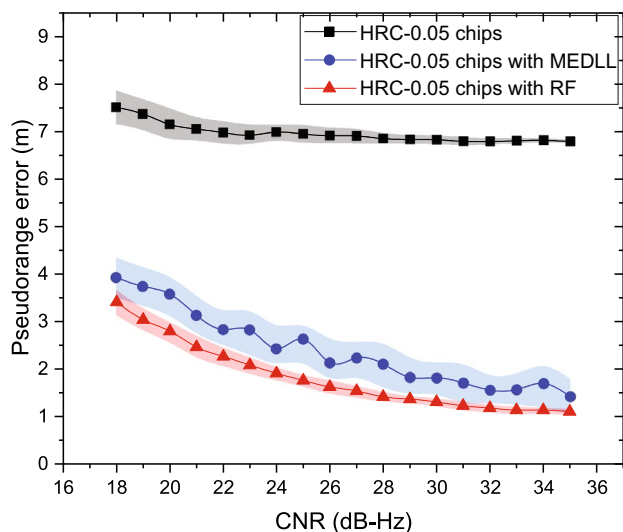

Fig. 16 Pseudorange errors at different CNRs when 0.05 chips spacing HRC is used for discrimination

more susceptible to random noise and less robust than the RF-based estimator in the multiple multipaths case. We speculate that this is also the reason why there are localized extreme points in the pseudorange error distributions of MEDLL in Figs. 11 and 12.

Overall, the performance of MEDLL is also much affected by the front-end bandwidth of received signals in the multiple-multipath case. The RF-based estimator demonstrates better multipath mitigation performance than MEDLL and no multipath correction case when the 0.1 chips spacing narrow correlator and the 0.05 chips spacing HRC are used for the discrimination, whereas MEDLL performs better using 1 chip spacing NEML discriminator.

MEDLL is less robust in the multiple-multipath case compared to the RF-based estimator and no multipath correction.

Discussion

The RF-based multipath parameter estimator takes equally spaced sample points of the autocorrelation function as the inputs. The sample points are obtained through a multi-correlator, and the sample spacing determines the number of correlators used. However, it is difficult to implement a large number of correlators in a receiver. Here, a feasibility evaluation is carried out to test the performance of the RF-based estimator with different spaced samples as the inputs. Ten tests are conducted to evaluate the performance of the RF-based estimator and compare with MEDLL. The sampling spacing settings and the corresponding number of correlators required in different tests are shown in Table 3. The parameters of samples in each training and testing dataset are the same as those in the one-multipath case, except the number and the sampling spacing of the inputs. The CNR and front-end bandwidth of received signals are set to 35 dB-Hz and 20 MHz, respectively. The RMSE of relative amplitude estimates and relative code delay estimates with different sampling spacing setting are shown in Fig. 17.

As shown in Fig. 17, the parameter estimation error obtained by the RF-based estimator always remains at the same level when the sampling chips spacing is no larger than 0.3 chips; when the sampling spacing is larger than 0.3 chips, the estimation error of code delay rises slightly with increasing sample spacing. Whereas the estimation error of both relative amplitude and code delay achieved by MEDLL increases sharply with increasing chip spacing setting. The results show that the RF-based estimator is rarely affected by the sampling spacing of the inputs. It demonstrates good multipath mitigation performance even when only a limited number of correlators are used. According to the results, a sampling spacing

Table 3 Sampling spacing settings of the multi-correlator and the corresponding number of correlators required

Test no	Sampling space	Number of correlators required
1	0.05	121
2	0.1	61
3	0.15	41
4	0.2	31
5	0.25	25
6	0.3	21
7	0.35	17
8	0.4	15
9	0.45	13
10	0.5	13
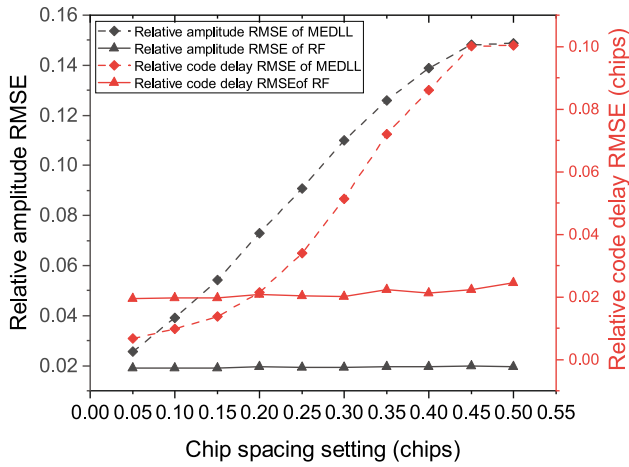
Fig. 17 RMSEs of multipath parameter estimates for inputs with different sampling spacing

Fig. 18 Skyplot mask of visible GPS satellites at the data collection site

of 0.3 chips would be a good choice to balance the feasibility and performance of the RF-based estimator. The number of correlators required at this point is 21.

Experiments

The above simulations are performed under ideal assumptions, i.e., the noise in the received signals is assumed to be AWGN and the autocorrelation function do not distort except for the effect of multipath. However, the noise in the real data is not exactly AWGN, and the shape of the autocorrelation function might be unpredictably deformed due to the presence of various disturbances. In order to verify whether the proposed RF-based parameter estimator can still work effectively in real urban environments, experiments on real data are conducted. The experimental setup is as follows: the data file we used is collected at 20 Msps using an USRP N210 in an urban area of Hong Kong (Tsim Sha Tsui East, 21/12/2022). Figure 18 shows the skyplot mask of visible Global Positioning System (GPS) satellites. The front-end bandwidth of the received signal is 2 MHz. The average CNR is at 39 dB-Hz.

Data preprocessing

Different from the signal model in (1), the GNSS signal model of real data is represented as

$$r_\Sigma(t) = \sqrt{2P} \sum_{i=0}^{M} a_i d(t - \tau_i) p(t - \tau_i) \cos\left(2\pi f_{IF} t + \phi_i\right) + n(t)$$

(17)

where P is the signal power, $d(\cdot)$ is the navigation message.
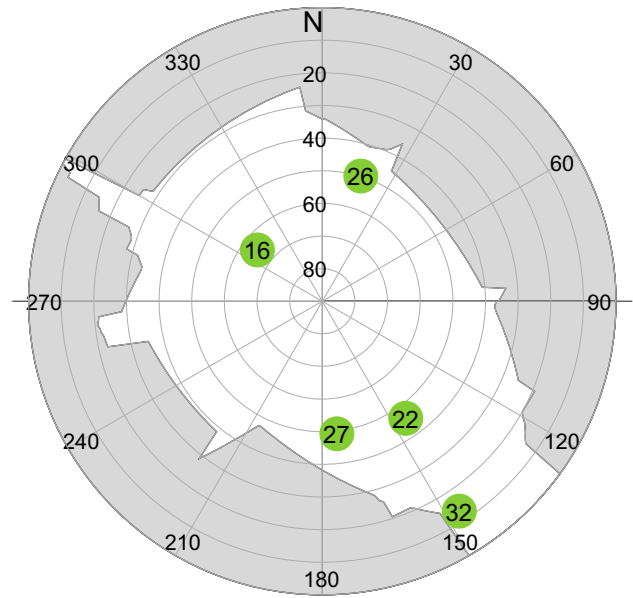
Correlating the received signal with local replica yields

$$R_\Sigma(t) = \sqrt{2P} \sum_{i=0}^{M} a_i d(t - \tau_i) R(t - \tau_i) e^{j\phi_i} + N(t)$$

(18)

At this time, $R_{\Sigma I}(t)$ is represented as

$$R_{\Sigma I}(t) = \sqrt{2P} d(t - \tau_0) \left\{ R(t - \tau_0) + \sum_{i=1}^{M} \alpha_{Ii} R(t - \tau_i) \right\} + N_I(t)$$

(19)

where

$$\alpha_{Ii} = \frac{d(t - \tau_i)}{d(t - \tau_0)} a_i \cos \overline{\phi}_i$$

(20)

Due to the uncertainties in signal power and the presence of navigation messages, the relative amplitude of reflected signal components with respect to the reference autocorrelation function may not be in the range of (0, 1). To make the relative amplitude of the reflected signal fall in (0, 1) so that the RF-based estimator works properly, the value of $\sqrt{2P}$ must satisfy $0 < \sqrt{2P} < 1$. However, the above condition is not always satisfied in real data. We preprocess the original autocorrelation function, let

$$R'_{\Sigma I}(t) = \beta R_{\Sigma I}(t)$$

(21)

where

$$\beta = 0.2 \frac{\max(R_{ref})}{\max(|R_{\Sigma I}(t)|)}$$

(22)

Considering the effect of navigation message on the autocorrelation function can be eliminated by multiplying ±1, if

Table 4 2D positioning statistical results

Type of DLL	Root mean squared error (m)	Mean absolute error (m)	Standard deviation (m)
Conventional DLL	12.4053	10.8567	6.0187
MEDLL	12.8385	10.4673	7.4545
DLL with RF-based estimator	11.3443	10.1133	5.1537

Table 5 3D positioning statistical results

Type of DLL	Root mean squared error (m)	Mean absolute error (m)	Standard deviation (m)
Conventional DLL	55.8687	49.4108	26.4501
MEDLL	62.1144	47.2343	40.4501
DLL with RF-based estimator	50.9879	45.3406	23.3888

noise is not concerned, the autocorrelation function of DP signal can be restored by

$$R(t - \tau_0) = R'_{\Sigma I}(t) - (P' - 1)R(t - \tau_{M+1}) - \sum_{i=1}^{M} \alpha'_{Ii} R(t - \tau_i) \tag{23}$$

where τ_{M+1} satisfy $|\tau_{M+1} - \tau_0| \to 0$; $P' = \beta\sqrt{2P}$; $\alpha'_{Ii} = \beta\sqrt{2P}\alpha_{Ii}$.

The multipath parameters to be estimated at time k are $\theta^k = \{P', \alpha'_{I1}, \alpha'_{I2}, \cdots, \alpha'_{IM}, \tau_{M+1}, \tau_1, \cdots, \tau_M\}$.

The estimates of P' serves two purposes: (1) to restore the amplitude of $R(t - \tau_0)$ to unity respected to R_{ref}, and (2) to calculate the relative amplitude estimates of multipaths. The former one makes no effect on multipath mitigation, and conversely, if there is an estimation error in τ_{M+1}, it will bring in a new error source in code discrimination. So $(P' - 1)R(t - \tau_{M+1})$ is ignored in the restoration of DP signal. $R(t - \tau_0)$ is finally restored by

$$R(t - \tau_0) = R'_{\Sigma I}(t) - \sum_{i=1}^{M} \hat{\alpha}'_{Ii} R(t - \hat{\tau}_i) \tag{24}$$

where $\hat{\alpha}_{Ii}, \hat{\tau}_i \in \hat{\theta}^k$ is the parameter estimates of i-th multipath at time k.

The amplitude of the restored $R(t - \tau_0)$ may be smaller than the original autocorrelation function, but this does not affect the discrimination results.

Experimental results

The preprocessed data is then processed using the open source SDR with conventional DLL, MEDLL and the DLL with RF-based estimator for code tracking, respectively. Both the MEDLL and the RF-based estimator use the three multipath signal model. The filter bandwidth of DLLs are set to 2 Hz. Here, the 1 chip spacing NEML discriminator is employed for code discrimination. The statistical results of 2D and 3D positioning over 180 epochs are shown in Tables 4 and 5.

According to the positioning results, the SDR using the DLL with RF-based estimator obtains lower positioning error than those using MEDLL and conventional DLL, and it achieves 8.5% lower RMSE for 2D positioning and 8.7% lower RMSE for 3D positioning than the SDR with conventional DLL over 180 epochs. Although the SDR with MEDLL obtained lower mean absolute error than that with conventional DLL, both the RMSE and the standard deviation are much larger than those of the SDR employing the DLL with RF-based estimator or conventional DLL.

The RF-based parameter estimator still performs well for real data collected in urban environment with the front-end bandwidth of 2 MHz. Although the SDR employing MEDLL achieves lower mean absolute error than the SDR with conventional DLL in this experiment, the larger RMSE and standard deviation indicate the poor robustness of MEDLL compared to the RF-based parameter estimator.

Computational complexity

In this subsection, we present an analysis on the computational complexity of the RF-based estimator and MEDLL. To ensure fairness in performance comparison, both the MEDLL and the RF-based estimators use 21 correlators from -3 to $+3$ chip in the above simulations and experiments, so the cost of correlation computation is the same for both. In the following, we test the time complexity of parameter estimation procedure for the RF-based estimator and MEDLL.

First, we test the time cost of a single parameter estimation with the MEDLL and the RF-based estimator. The time costs are tested on a desktop with an Intel Core i9-12900 K 24-Core Processor CPU@3.19GHZ and 32.00 GB memory. The average time cost for 100 runs of both methods is shown in Table 6. As shown in Table 6, the RF-based parameter estimator spends much less time on parameter estimation

Table 6 Average time cost of MEDLL and RF-based estimator for 100 runs

Method	Data pre-processing (ms)	Parameter estimation (ms)	Autocorrelation function correction (ms)	Total (ms)
MEDLL	–	137.184334	1.479606	138.663940
RF-based estimator	0.052164	2.859061	0.066277	2.977502

than MEDLL. This is due to the fact that the decision trees in random forest make predictions in parallel, thus effectively reducing the runtime. MEDLL also takes more time in autocorrelation function correction because it finds the signal with the shortest delay as the LOS signal before performing correction.

The training of random forest is also time consuming, but since the training of the model can be completed offline, it does not affect the real-time performance. As for model calling, in practice the model only needs to be read once before use, so it is not considered. The random forest model is stored in memory in use after reading, which needs more memory than MEDLL.

We also theoretically analyze the time complexity of MEDLL and RF-based parameter estimator. The focus here is on the trend of time complexity as the number of multipaths increases. In the worst case, the time complexity of random forest building and prediction is $O\left(N_T K \tilde{N}^2 \log^2 \tilde{N}\right)$ and $O\left(N_T N\right)$, respectively, where N_T denotes the number of decision trees, N denotes the number of samples in the training set and K is the number of variables randomly drawn at each node. $\tilde{N} = 0.632N$, due to the fact that bootstrap samples draw, on average, 63.2% of unique samples (Louppe 2014). The complexity of multiple output decision tree is the same as single output decision tree (Pliakos et al. 2018). Thus, the RF-based parameter estimator remains constant time complexity as the number of multipaths increases. As for MEDLL, its time complexity is $O\left(T^M\right)$, where T is the number of iterations and M is the number of multipaths. Therefore, the runtime of MEDLL will increase exponentially with the number of multipaths.

Conclusion

This study proposes a random forest-based multipath parameter estimator which can estimate and mitigate the multipath for GNSS signal in urban environments. On one hand, the RF-based estimator demonstrates better robustness compared to MEDLL — its multipath mitigation performance is rarely affected by the front-end bandwidth setting of received signals and noise randomness. The simulation results show that the RF-based estimator exhibits much better multipath mitigation performance than MEDLL for signals with front-end bandwidths below 6 MHz. In the experiments for the real data with a front-end bandwidth of 2 MHz, the SDR employing the DLL with RF-based estimator reduces the 2D and 3D positioning RMSE by 8.5% and 8.7% over 180 epochs compared to the SDR with conventional DLL, whereas MEDLL does not show superiority. On the other hand, compared to the machine learning-based methods mentioned before, the RF-based estimator focuses on multipath parameter

estimation rather than correlator or discriminator design. And it requires only 21 correlators in total, which shows great feasibility in receivers. However, our current algorithm just addresses the multipath mitigation problem for pseudorange measurements. The multipath mitigation for carrier phase remains to be explored.

Declarations

Conflict of interest The authors declare no competing interests.

Consent for publication All authors read and approved the final manuscript

References

Aissou G, Benouadah S, El Alami H, Kaabouch N (2022) Instance-based supervised machine learning models for detecting GPS spoofing attacks on UAS. In: 2022 IEEE 12th annual computing and communication workshop and conference (CCWC), 0208–0214.

Anand B, Senapati M, Barsaiyan V, Rajalakshmi P (2021) LiDAR-INS/GNSS-based real-time ground removal, segmentation, and georeferencing framework for smart transportation. IEEE Trans Instrum Meas 70:1–11

Biau G, Scornet E (2016) A random forest guided tour. TEST 25:197–227

Borhani-Darian P, Li H, Wu P, Closas P (2023) Deep learning of GNSS acquisition. Sensors 23(3):1566

Borre K, Akos DM, Bertelsen N, Rinder P, Jensen SH (2007). A software-defined GPS and Galileo receiver: a single-frequency approach. Springer Science Business Media.

Breiman L (1996) Bagging predictors. Mach Learn 24:123–140

Breiman L (2001) Random forests. Mach Learn 45:5–32

Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and regression trees. Wadsworth and Brooks/Cole: Monterey, CA, USA.

de Miguel MA, García F, Armingol JM, (2020) Improved LiDAR probabilistic localization for autonomous vehicles using GNSS. Sensors 20(11):3145

Du Y, Saha SS, Sandha SS, Lovekin A, Wu J, et al (2023) Neural-kalman gnss/ins navigation for precision agriculture. In: International conference on robotics and automation (ICRA).

Fenton PC, Jones J (2005) The theory and performance of NovAtel Inc.'s vision correlator. In: Proc. ION GNSS 2005, Institute of navigation, long beach, CA, USA, 2178–2186.

Garin L (1996) Strobe & edge correlator multipath mitigation for code. In: Proc. ION GPS 1996, Institute of navigation, Kansas City, MO, USA, 657–664.

Granger R, Simpson S (2008) An analysis of multipath mitigation techniques suitable for geodetic antennas. In: Proc. ION GNSS 2008, Institute of Navigation, Savannah, GA, USA, 2755–2765.

Hsu L-T (2017) GNSS multipath detection using a machine learning approach. In: 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), pp 1–6.

Kanhere AV, Gupta S, Shetty A, Gao G (2022) Improving gnss positioning using neural-network-based corrections. Navigation 69(4):548. https://doi.org/10.33012/navi.548

Lee YC (2002) Compatibility of the new military GPS signals with non-aviation receivers. In: Proceedings of the 58th annual meeting of the institute of navigation and CIGTF 21st guidance test symposium, pp 581–597.

Li H, Borhani-Darian P, Wu P, Closas P (2022) Deep neural network correlators for GNSS multipath mitigation. IEEE Trans Aerosp Electron Sys 59(2):1249–1259

Louppe G (2014) Understanding random forests: from theory to practice. PhD thesis. University of Liège.

McGraw GA, Braasch MS (1999) GNSS multipath mitigation using gated and high resolution correlator concepts. In: Proceedings of the 1999 national technical meeting of the institute of navigation, pp 333–342.

Van Nee RD, Siereveld J, Fenton PC, Townsend B R (1994) The multipath estimating delay lock loop: approaching theoretical accuracy limits. In: Proc. IEEE/ION PLANS 1994, IEEE, Las Vegas, NV, USA, April 11–15, 246–251.

Orabi M, Khalife J, Abdallah AA, Kassas ZM, Saab SS (2020) A machine learning approach for GPS code phase estimation in multipath environments. In: Proc. IEEE/ION PLANS 2020, institute of navigation, Portland, Oregon, USA, April 20–23, 1224–1229.

Phan Q-H, Tan S-L, McLoughlin I (2013) GPS multipath mitigation: a nonlinear regression approach. GPS Solut 17:371–380

Pliakos K, Pierre G, Celine V (2018) Global multi-output decision trees for interaction prediction. Mach Learn 107:1257–1281

Qi X, Xu B (2023) Machine learning assisted multipath signal parameter estimation and its evaluation under weak signal environment. In: Proc. IEEE/ION PLANS 2023, institute of navigation, Monterey, California, USA, April 24–27, 1019–1026.

Quan Y, Lau L, Roberts GW, Meng X, Zhang C (2018) Convolutional neural network based multipath detection method for static and kinematic GPS high precision positioning. Remote Sens 10(12):2052

Savas C, Dovis F (2019) Multipath Detection based on K-means clustering. In: Proc. ION GNSS+ 2019, institute of navigation, Miami, Florida, USA, 3801–3811.

Semanjski S, Muls A, Semanjski I, De Wilde W (2019) Use and validation of supervised machine learning approach for detection of GNSS signal spoofing. In: 2019 international conference on localization and GNSS (ICL-GNSS), pp 1–6.

Suzuki T, Kusama K, Amano Y (2020) NLOS multipath detection using convolutional neural network. In: Proc. ION GNSS+ 2020, institute of navigation, Virtual, 2989–3000.

Townsend B, Fenton P (1994) A practical approach to the reduction of pseudorange multipath errors in a L1 GPS receiver. In: Proc. ION GPS 1994, institute of navigation, salt lake city, Utah, USA, 143–148.

Van Dierendonck A, Fenton P, Ford T (1992) Theory and performance of narrow correlator spacing in a GPS receiver. Navigation 39(3):265–283

Wang Z, Chen W, Dong D, Wang M, Cai M et al (2019) Multipath mitigation based on trend surface analysis applied to dual-antenna receiver with common clock. GPS Solut 23:1–15

Weill LR (2002) Multipath mitigation using modernized GPS signals: how good can it get?. In: Proc. ION GPS 2002, institute of navigation, Portland, OR, USA, 493–505.

Wu J, Dempster AG (2012) Tailored spreading symbols in "Strobe Correlators" using "W2-pulses." Signal Process 92(5):1189–1201

Xie G (2009) Principles of GPS and receiver design. Publishing House of Electronics Industry, Beijing 7: 61–63

Xie L, Cui X, Zhao S, Lu M (2017) Mitigating multipath bias using a dual-polarization antenna: Theoretical performance, algorithm design, and simulation. Sensors 17(2):359

Zhang G, Ng H-F, Wen W, Hsu L-T (2020) 3D mapping database aided GNSS based collaborative positioning using factor graph optimization. IEEE Trans Intell Transp Syst 22(10):6175–6187

Xin Qi received her B.S. degree in automation from Zhengzhou University in 2019 and M.S. degree in control science and engineering from Harbin Institute of Technology in 2021. She is studying as a PhD student in the Department of Aeronautical and Aviation Engineering at The Hong Kong Polytechnic University. Her research interests include machine learning-assisted GNSS signal processing, urban navigation and positioning.

Bing Xu received his BEng and Ph.D. degrees in network engineering and navigation guidance and control from Nanjing University of Science and Technology, Nanjing, China, in 2012 and 2018, respectively. He is currently an Assistant Professor with the Department of Aeronautical and Aviation Engineering at The Hong Kong Polytechnic University. His current research interests include GNSS signal processing in software-defined receivers.

Zhipeng Wang received his bachelor's degree in Communication Engineering from Northwestern Polytechnical University (NWPU) in 2006 and his Ph.D. degree in Trafc Information Engineering from Beihang University in 2013. He is now a professor at the Department of Electronic Information Engineering, Beihang University, China. His research focuses on high precision aviation navigation Minimum Operation Network (MON), aviation navigation integrity monitoring, GNSS aviation application test verification and international standardization.

Li-Ta Hsu received his B.S. and Ph.D. degrees in aeronautics and astronautics from National Cheng Kung University, Tainan, Taiwan, in 2007 and 2013, respectively. He is currently an Associate Professor with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University. His current research interests include global navigation satellite system positioning in challenging environments and localization for autonomous driving vehicles and unmanned aerial vehicles.