AGU PUBLICATIONS

Radio Science

RESEARCH ARTICLE
10.1002/2015RS005664

**Key Points:**
- A new multi-PLL architecture to increase GPS receiver reliability
- A two-stage fusion algorithm to filter the GPS Doppler measurements
- Enhanced GPS receiver robustness and accuracy under scintillations

**Correspondence to:**
Z. Liu,
lszzliu@polyu.edu.hk

**Citation:**
Xu, R., Z. Liu, and W. Chen (2015), Multi-PLL with two-stage fusion to mitigate ionospheric scintillation effects on GPS receivers, *Radio Sci.*, *50*, 630–641, doi:10.1002/2015RS005664.

Received 29 JAN 2015
Accepted 28 MAY 2015
Accepted article online 2 JUN 2015
Published online 6 JUL 2015



# Multi-PLL with two-stage fusion to mitigate ionospheric scintillation effects on GPS receivers

**Rui Xu[1], Zhizhao Liu[1], and Wu Chen[1]**

[1]Department of Land Surveying and Geo-Informatics, Hong Kong Polytechnic University, Hong Kong, China

**Abstract** Ionospheric scintillation poses a great threat to the reliability and accuracy of Global Positioning System (GPS) in various applications. It can increase tracking errors of the phase-locked loop (PLL) in a GPS receiver and even cause the PLL loss of lock under severe scintillations. To mitigate the effect of scintillation on GPS receivers, especially to reduce the occurrence of loss of lock, a multi-PLL with two-stage fusion (i.e., tracking fusion and output fusion) is proposed in this paper. This algorithm integrates several parallel sub-PLLs with different loop parameters into one channel to track one GPS satellite's signal. Every sub-PLL has its own discriminator, loop filter, carrier numerical controlled oscillator, and a tracking fusion (i.e., the first stage fusion). The tracking fusion of each sub-PLL integrates the Doppler frequency measurements from all other sub-PLLs to detect the state of its own sub-PLL and feeds back reliable Doppler frequency measurements. Simultaneously, the tracking fusion outputs the Doppler frequency measurements to the second stage fusion (i.e., output fusion), which integrates the outputs from all tracking fusions to provide continuous and accurate Doppler frequency measurements for the following positioning/navigation estimator. Performances of the proposed algorithm are tested using real-world GPS data with different levels of scintillations and compared with results from single-PLLs. For three real-world scintillation cases ($S_4 = 0.26–1.1$, $\sigma_\phi = 0.05–1.49$ rad, and average $C/N_0 = 41.2–45.7$ dB Hz), the multi-PLL algorithm performs more robustly than the single-PLLs and is able to keep tracking in all scintillation cases.

## 1. Introduction

Global Positioning System (GPS) signals are affected by various error sources when they travel from satellites to users. One of the most significant error sources is the ionosphere. The ionosphere often has two types of effect on the GPS signals: one is ionospheric range error and the other is ionospheric scintillation. The ionospheric range error causes a range delay to GPS pseudorange measurements and a phase advance to carrier phase measurements. This ionospheric range error can be considerably reduced by combining measurements from GPS dual frequencies, performing differential GPS or adopting ionospheric delay models [*Gao et al.*, 2002; *Xu*, 2007; *Luo et al.*, 2014]. However, the ionospheric scintillation effect is difficult to compensate. Ionospheric scintillation occurs when GPS signals pass though ionospheric electrons with irregular density distribution. It can lead to rapid phase and amplitude fluctuations in GPS signals. The rapid phase fluctuation in the GPS signal is known as phase scintillation, often characterized by an index $\sigma_\phi$ [*Leick*, 2004; *Humphreys*, 2005]. The rapid amplitude fluctuation is called as amplitude scintillation, usually characterized by an index $S_4$ [*Leick*, 2004; *Humphreys*, 2005]. Both phase and amplitude scintillations can result in cycle slips and loss of lock in a GPS receiver [*Crane*, 1977; *Van Dierendonck et al.*, 1993; *Morrissey et al.*, 2004; *Chiou et al.*, 2008; *Humphreys et al.*, 2010; *De Oliveira Moraes et al.*, 2011; *Xu et al.*, 2014]. The phase scintillation can yield an additional Doppler shift in the GPS signal and increase the total Doppler shift. When the total Doppler shift exceeds the bandwidth of carrier tracking loop, loss of phase lock will probably occur [*Chiou et al.*, 2008]. The amplitude scintillation degrades the carrier-to-noise ($C/N_0$) ratio of GPS signal. When $C/N_0$ reduces to a level below the receiver's tracking threshold, the PLL tends to lose the signal [*Van Dierendonck et al.*, 1993; *De Oliveira Moraes et al.*, 2011]. When multiple satellites experience loss of lock simultaneously, GPS positioning and navigation outage may occur.

To reduce the occurrence frequency of loss of lock due to ionospheric scintillations, it is necessary to improve the robustness of carrier tracking loop of GPS receivers. Carrier tracking loop is commonly realized using a traditional phase-locked loop (PLL) that generally consists of a discriminator, a low-pass loop filter, and a numerical controlled oscillator (NCO). However, its tracking performance under ionospheric scintillation largely depends on the loop parameters, such as bandwidth and integration time [*Zhang and Morton*, 2009;

*Zhang et al.*, 2010]. With lower bandwidth (3–5 Hz) and longer integration time (5–20 ms), the PLL can keep signal tracking under stronger amplitude scintillations ($S_4 > 1.0$) [*Skone et al.*, 2005; *Forte*, 2012]. However, with a low bandwidth ≤10 Hz, maintaining PLL tracking under strong phase scintillation ($S_4 = 0.175$ and $\sigma_\phi = 1.15$ rad) is difficult [*Skone et al.*, 2005]. In this case ($S_4 = 0.175$ and $\sigma_\phi = 1.15$ rad), PLL with a high bandwidth (>10 Hz) has the ability to track signals [*Skone et al.*, 2005]. The opposite requirements on the loop parameters (loop bandwidth and integration time) of amplitude and phase scintillations seriously affect the performance of PLL. Not only are GPS receivers with traditional PLL affected, but also those receivers are designed with advanced carrier tracking loops, such as an FLL-assisted PLL, Kalman filter-based phase-locked loop, and vector-based tracking algorithm [*Ganguly et al.*, 2004; *Humphreys*, 2005; *Chiou et al.*, 2007, 2008; *Zhang and Morton*, 2009; *Zhang et al.*, 2010; *Won et al.*, 2012; *Won and Eissfeller*, 2013; *Xu et al.*, 2014]. When proper loop bandwidth and integration time are selected, these algorithms show robust performance under scintillation conditions. For instance, FLL-assisted PLL with 1–2 Hz FLL bandwidth and 3–15 Hz PLL bandwidth is able to maintain signal tracking under strong amplitude scintillation ($C/N_0$ fading >30 dB Hz) [*Zhang et al.*, 2010]. Using optimal bandwidth is a solution to improve the robustness of carrier tracking loop under scintillations. However, fixing the optimal bandwidth under scintillation condition may limit the performance of the GPS receiver under other conditions, such as high dynamic condition. Therefore, the loop parameters must be appropriately adjusted to customize the environment variations. When scintillation occurs, they can be adjusted to accommodate the intensity of phase and amplitude scintillations. This can be achieved by using an adaptive algorithm, but it usually involves complex scintillation predetection and bandwidth updating algorithm [*Ganguly et al.*, 2004; *De Oliveira Moraes et al.*, 2011]. In addition, the reliability of single carrier tracking loop is limited [*Yeh and Hsieh*, 2011; *Won et al.*, 2012]. Once loss of lock occurs to a single tracking loop, the tracking loop must reacquire the signal. During strong scintillation period, the common signal acquisition may require a long time.

To mitigate scintillation effects by determining appropriate PLL loop parameters, in this study a multi-PLL carrier tracking algorithm is proposed. This algorithm integrates several parallel sub-PLLs of different loop parameters in a two-stage fusion. Specifically, each channel uses multiple PLLs to track the carrier signal from the same satellite. The first stage fusion is called tracking fusion, and it is implemented in every sub-PLL of each channel. It uses the Doppler frequency measurements from all the sub-PLLs to maintain its sub-PLL in a stable state. The second stage fusion is output fusion that is implemented in each channel and is responsible for outputting accurate and continuous Doppler frequency measurements to positioning/navigation estimator. The multi-PLL algorithm is tested using real-world GPS data with moderate or strong scintillations. In addition, the performance of the multi-PLL algorithm is compared with that of single-PLLs to show the improvement of this algorithm.

This paper is organized as follows. Section 2 presents the architecture of the traditional single-PLL and that of the proposed multi-PLL algorithm. Implementation of the multi-PLL with two-stage fusion algorithm is developed in section 3. In section 4, the tracking results of this multi-PLL algorithm under simulated and real-world GPS IF data collected under scintillation conditions are analyzed. The conclusions of this paper are given in section 5.

## 2. Architecture of the Multi-PLL Algorithm

This section describes the architecture of the proposed multi-PLL algorithm, which is preceded by a brief recall of the traditional single-PLL.

### 2.1. Traditional Single-PLL Architecture

For a traditional GPS receiver, the signal from one satellite is tracked in one channel and each channel contains only one PLL, as shown in Figure 1. The PLL consists of one discriminator, one loop filter, and one numerical control oscillator (NCO).

As shown in Figure 1, the PLL discriminator calculates the phase difference $\phi_e$ (i.e., phase error) between the received carrier and the local carrier. The phase error $\phi_e$ is inputted to the loop filter, which is commonly implemented by a first-order or second-order low-pass filter. The loop filter outputs filtered phase error $\phi_f$ to the carrier NCO. In a software-defined receiver, the carrier NCO estimates the Doppler frequency correction $\Delta\omega_D$ to correct rough carrier frequency $\omega_{\text{carr}}^{\text{acq}}$ from acquisition module. The integral of $\omega_{\text{carr}}^{\text{acq}}$ is

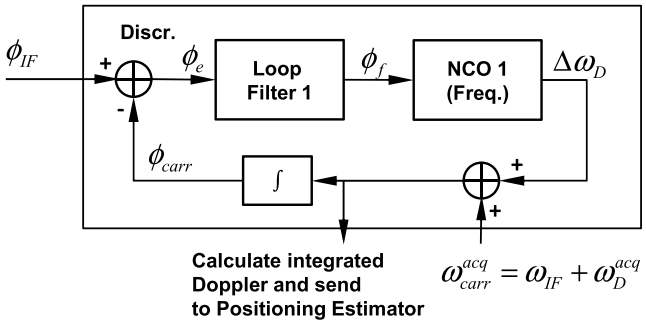


**Figure 1.** Architecture of a general PLL in one channel for software-defined receiver.

used to generate local carrier. And the Doppler frequency measurement $\omega_D$ is calculated by $\omega_{\text{carr}}^{\text{acq}} + \Delta\omega - \omega_{\text{IF}}$. The integral of the Doppler frequency measurement from all channels is outputted to positioning/navigation estimator to calculate user's position and velocity information.

### 2.2. The Multi-PLL Architecture

Different from the traditional single-PLL, the proposed multi-PLL uses an architecture with multiple redundant PLLs, as shown in Figure 2. It integrates several parallel PLLs (called sub-PLL) in one channel to track the carrier signal of one GPS satellite. In Figure 2, the multi-PLL contains *N* sub-PLLs. Each sub-PLL has its own discriminator, loop filter, and NCO to ensure its independence. These sub-PLLs collaborate with each other via a two-stage fusion in which the measurements from multiple sub-PLLs are combined. In the first stage, tracking fusion is implemented in each sub-PLL. It can detect the state of its own sub-PLL, feed back reliable Doppler frequency measurements, and output reliable Doppler frequency measurements to the second stage fusion (i.e., output fusion). Each channel has only one output fusion to integrate the Doppler frequency measurements obtained from all the sub-PLLs and generate an accurate result to positioning/navigation estimator.

In sub-PLL *n* ($n = 1, 2, \ldots, N$), PLL discriminator *n* measures the phase error $\phi_{e,n}$; loop filter *n* removes the noise of $\phi_{e,n}$; and NCO *n* generates a Doppler frequency measurement $\omega_{D,n}$ and outputs $\omega_{D,n}$ to tracking fusion *n*, as shown in Figure 2. The tracking fusion *n* collects and uses the Doppler frequency measurements from all the sub-PLLs and the predicted Doppler frequency $\bar{\omega}_{D,n}$ from an estimator to estimate a reliable Doppler frequency measurement $\hat{\omega}_{D,n}$. The $\hat{\omega}_{D,n}$ is fed back to sub-PLL *n* and simultaneously outputted to the output fusion, which integrates all the $\hat{\omega}_{D,n}$ from all the sub-PLLs to estimate the outputted Doppler frequency measurement $\hat{\omega}_D^{\text{output}}$. The $\hat{\omega}_D^{\text{output}}$ is then used to generate carrier phase measurement and at the same time update the $\bar{\omega}_D$ estimator.

## 3. Two-Stage Fusion Algorithm

The multi-PLL uses a two-stage fusion algorithm, which is to be described in this section, to complete carrier signal tracking and measurement output. While the tracking fusion aims to keep each sub-PLL in lock, the output fusion aims to provide continuous and accurate Doppler frequency measurement.

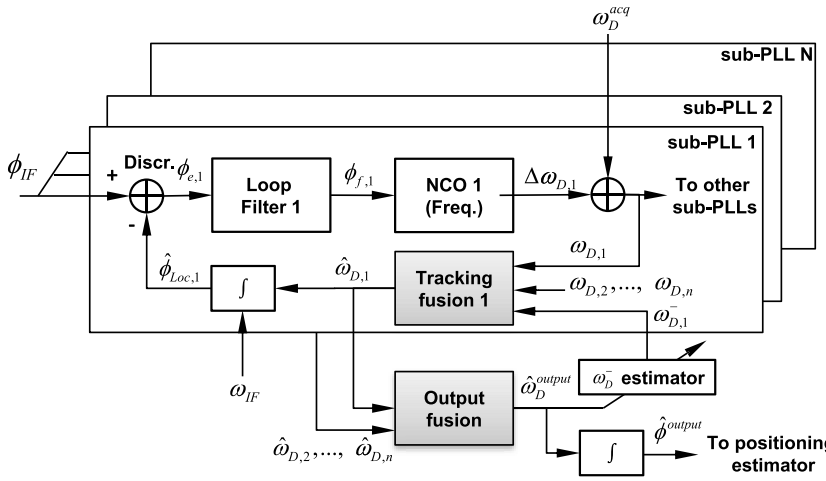


**Figure 2.** Architecture of the multi-PLL.

### 3.1. Tracking Fusion

Assuming that in the multi-PLL, there are $N$ sub-PLLs. In the $n$th sub-PLL, the tracking fusion $n$ uses Doppler frequency measurements $\omega_D = [\omega_{D,1}, \ldots, \omega_{D,n-1}, \omega_{D,n+1} \ldots, \omega_{D,N}]$ from other sub-PLLs and the predicted Doppler frequency $\bar{\omega}_{D,n}$ to detect the state of sub-PLL $n$ and feed back a reliable value $\hat{\omega}_{D,n}$. The Doppler frequency measurement $\omega_{D,n}$ without fusion in sub-PLL $n$ can be modeled as

$$\omega_{D,n}(k) = \omega_D^{\text{true}}(k) + v_n(k) \tag{1}$$

where $\omega_D^{\text{true}}(k)$ represents the true Doppler frequency at the $k$th moment and $v_n(k)$ represents the noise of sub-PLL $n$ which is modeled as Gaussian white noise $\left(v_n(k) \sim N(0, \sigma_n^2)\right)$. Therefore, the difference between the Doppler frequency measurements from any two sub-PLLs $n$ and $m$ can be written as

$$\varepsilon_{nm}(k) = \omega_{D,n}(k) - \omega_{D,m}(k) = v_n(k) - v_m(k), \quad (m = 1, \ldots, N \text{ and } m \neq n) \tag{2}$$

where $\varepsilon_{nm}(k)$ represents the difference of measured Doppler frequency between sub-PLL $n$ and sub-PLL $m$ at the $k$th moment. It is assumed that sub-PLLs $n$ and $m$ operate indepenently. Since the noise $v_n(k) \sim N(0, \sigma_n^2)$ and $v_m(k) \sim N(0, \sigma_m^2)$, $\varepsilon_{nm}(k)$ follows the Gaussian distribution too with $\varepsilon_{nm}(k) \sim N(0, \sigma_{nm}^2)$ where $\sigma_{nm}^2 = \sigma_n^2 + \sigma_m^2$. It should be noticed that sub-PLLs $n$ and $m$ probably have different integration times. Thus, a preprocessing is needed before tracking fusion to synchronize the outputs of different sub-PLLs. The preprocessing method is illustrated in Appendix A. To simplify the synchronization process, integration time of 1, 5, 10, or 20 ms is suggested for each single sub-PLL.

The closeness of the Doppler frequency measurements from two sub-PLLs is denoted by $\varepsilon_{nm}(k)$. Since sub-PLLs $n$ and $m$ measure the Doppler frequency of one satellite, sub-PLLs $n$ and $m$ should provide similar results after the synchronization processing and their difference $\varepsilon_{nm}(k)$ should be within $\pm 3\sigma_{nm}$ with a high confidence coefficient (99.7%) under normal conditions. In this case, the two sub-PLLs are said to be compatible with each other. If one or both of them output incorrect Doppler frequency measurements, the value of $\varepsilon_{nm}(k)$ is likely to fall out of the range $\pm 3\sigma_{nm}$. Thus, a degree of compatibility is defined as

$$\gamma_{nm}(k) = \begin{cases} 1 - \dfrac{|\varepsilon_{nm}(k)|}{3\sigma_{nm}(k-1)}, & |\varepsilon_{nm}(k)| < 3\sigma_{nm}(k-1) \\ 0, & |\varepsilon_{nm}(k)| \geq 3\sigma_{nm}(k-1) \end{cases} \tag{3}$$

Obviously, $\gamma_{nm}(k)$ ranges from 0 to 1. If $\gamma_{nm}(k) > 0$, sub-PLL $n$ is said to be compatible with sub-PLL $m$. If sub-PLL $n$ is compatible with more than $\lceil N/3 \rceil$ PLLs, sub-PLL $n$ is considered stable and its outputs are considered reliable. If $\gamma_{nm}(k) = 0$, sub-PLLs $n$ and $m$ are said incompatible with each other, primarily because one or both of them output incorrect Doppler frequency measurements. If sub-PLL $n$ is incompatible with more than $\lceil 2N/3 \rceil$ PLLs, the state of sub-PLL $n$ will be discussed in the next paragraph.

To deal with the situation where sub-PLL $n$ is incompatible with more than $\lceil 2N/3 \rceil$ PLLs, another index $\varepsilon_{nn}(k)$ is used to measure the error of $\omega_{D,n}(k)$ from sub-PLL $n$. The $\varepsilon_{nn}(k)$ is defined as

$$\varepsilon_{nn}(k) = \omega_{D,n}(k) - \bar{\omega}_{D,n}(k) = v_n(k) \tag{4}$$

where $\bar{\omega}_{D,n}(k)$ is the predicted Doppler frequency measurement synchronized to the output of sub-PLL $n$ at the moment $k$. Here it is assumed that the predicted $\bar{\omega}_{D,n}(k)$ can approximately represent $\omega_D^{\text{true}}(k)$ and the residual error of this representation is ignored. $\bar{\omega}_{D,n}(k)$ is obtained from a $\bar{\omega}_D$ estimator, which is a low-order polynomial and can be written as

$$\begin{aligned} &\bar{\omega}_{D,n}(k) = a_{k_{\text{est}}} + b_{k_{\text{est}}} k T_{\text{coh},n} + c_{k_{\text{est}}} \left(k T_{\text{coh},n}\right)^2 \\ &k_{\text{est}} T_{\text{est}} < k T_{\text{coh},n} \leq (k_{\text{est}} + 1) T_{\text{est}} \end{aligned} \tag{5}$$

where $\bar{\omega}_{D,n}(k)$ represents the $k$th predicted Doppler frequency measurement in sub-PLL $n$ at the $kT_{\text{coh},n}$ moment, which is between two successive updating moments, i.e., $k_{\text{est}} T_{\text{est}} < k T_{\text{coh},n} \leq (k_{\text{est}} + 1) T_{\text{est}}$; $T_{\text{coh},n}$ represents the integration time of sub-PLL $n$; $a_{k_{\text{est}}}$, $b_{k_{\text{est}}}$, and $c_{k_{\text{est}}}$ are coefficients of $\bar{\omega}_D$ estimator which are updated every $T_{\text{est}}$; and $k_{\text{est}}$ represents the number of times $\bar{\omega}_D$ estimator has been updated.

In (5), the coefficients $a_{k_{\text{est}}}$, $b_{k_{\text{est}}}$, and $c_{k_{\text{est}}}$ are updated using $\hat{\omega}_D^{\text{output}}$ collected from output fusion (Figure 2). If $\hat{\omega}_D^{\text{output}}$ is outputted at a $T_{\text{out}}$ sampling period, the number of samples of $\hat{\omega}_D^{\text{output}}$ collected within one

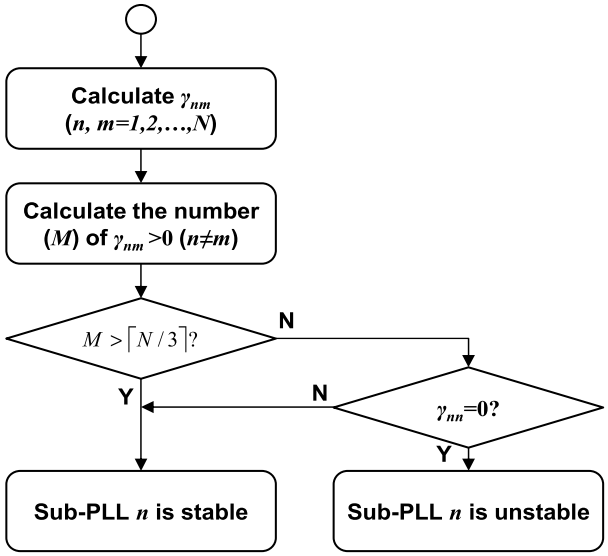


**Figure 3.** Flowchart of detecting sub-PLL $n$ state detection.

coefficient update interval $T_{\text{est}}$ is $M = T_{\text{est}}/T_{\text{out}}$. Then, the coefficients $a_{k_{\text{est}}}$, $b_{k_{\text{est}}}$, and $c_{k_{\text{est}}}$ can be estimated by solving the equation:

$$(a_{k_{\text{est}}}, b_{k_{\text{est}}}, c_{k_{\text{est}}}) = \underset{(a_{k_{\text{est}}},\ b_{k_{\text{est}}},\ c_{k_{\text{est}}})}{\arg\ \min} \sum_{k=(k_{\text{est}}-1)M}^{k_{\text{est}}M-1} \left[a_{k_{\text{est}}} + b_{k_{\text{est}}} kT_{\text{out}} + c_{k_{\text{est}}}(kT_{\text{out}})^2 - \hat{\omega}_D^{\text{output}}(k)\right]^2 \tag{6}$$

To reduce the computational load, the update interval $T_{\text{est}}$ is usually set to be much larger than the coherent integration time of sub-PLL $n$ ($T_{\text{est}} \gg T_{\text{coh},n}$). However, the update interval affects the selection of the order of the $\omega_D^-$ estimator. For large update interval $T_{\text{est}}$, the $\omega_D^-$ estimator should select a high order to follow the Doppler frequency variations. Otherwise, a lower order can be selected. In this study, the update interval is 1 s and the order of the $\omega_D^-$ estimator is two, as shown in (5).

During the update interval $T_{\text{est}}$, $\omega_{D,n}^-(k)$ is considered unbiased. Hence, $\varepsilon_{nn}(k)$ as shown in (4) can be considered as measurement noise of sub-PLL $n$ $\left(\varepsilon_{nn}(k) \sim N\left(0, \sigma_{v,n}^2\right)\right)$. Then, the $\gamma_{nn}(k)$ is defined as

$$\gamma_{nn}(k) = \begin{cases} 1 - \dfrac{|\varepsilon_{nn}(k)|}{3\sigma_n(k-1)}, & |\varepsilon_{nn}(k)| < 3\sigma_n(k-1) \\ 0, & |\varepsilon_{nn}(k)| \geq 3\sigma_n(k-1) \end{cases} \tag{7}$$

Similarly, $\gamma_{nn}(k)$ ranges from 0 to 1. When $\gamma_{nn}(k) = 0$, sub-PLL $n$ is considered unstable and incompatible with the $\omega_D^-$ estimator. If $\gamma_{nn}(k) > 0$, sub-PLL $n$ is considered stable and compatible with the $\omega_D^-$ estimator.

Both $\gamma_{nm}(k)$ and $\gamma_{nn}(k)$ are used to detect the state of sub-PLL $n$, as illustrated in Figure 3. In sub-PLL $n$, the tracking fusion calculates $\gamma_{nm}(k)$ ($n \neq m$) at the $k$th moment and counts the number of cases with $\gamma_{nm} > 0$. If the number is larger than $\lceil N/3 \rceil$, sub-PLL $n$ is considered stable. Otherwise, the $\gamma_{nn}(k)$ is further calculated in the tracking fusion. If $\gamma_{nn} > 0$, sub-PLL $n$ is considered stable. Otherwise, it is considered unstable.

According to the state of sub-PLL $n$ (stable or unstable), tracking fusion $n$ feeds back Doppler frequency measurements as follows:

if $M > \lceil N/3 \rceil$
  sub-PLL is stable; $\hat{\omega}_{D,n}(k) \leftarrow \omega_{D,n}(k)$
else if $\gamma_{nn} > 0$
  sub-PLL is stable; $\hat{\omega}_{D,n}(k) \leftarrow \omega_{D,n}(k)$
else if $\gamma_{ii} = \max\{\gamma_{11}, \gamma_{22}, \ldots, \gamma_{NN}\}$ and $\gamma_{ii} > 0$
  sub-PLL is unstable; $\hat{\omega}_{D,n}(k) \leftarrow \omega_{D,i}(k)$

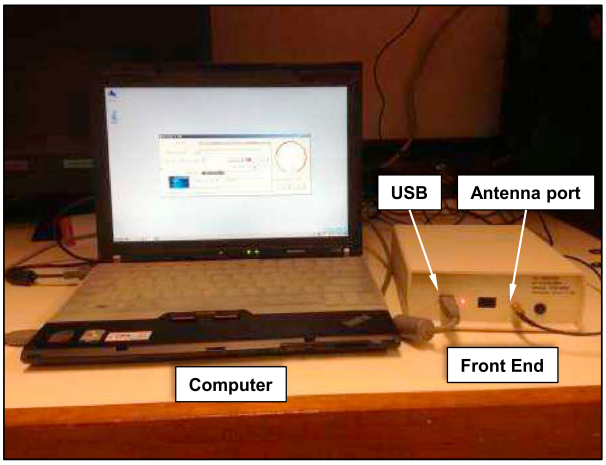


**Figure 4.** Real GPS IF data collection system.

else if $\gamma_{11} = \gamma_{22} = \ldots = \gamma_{NN} = 0$
sub-PLL is unstable; $\hat{\omega}_{D,n}(k) \leftarrow \bar{\omega}_{D,n}(k)$

As shown in (8), if sub-PLL $n$ is stable, the tracking fusion outputs its own Doppler frequency measurement $\omega_{D,n}(k)$. Otherwise, if sub-PLL $n$ is unstable, the output of tracking fusion is obtained from sub-PLL $i$ that has the largest $\gamma_{ii}$ among the other sub-PLLs. Meanwhile, sub-PLL $n$ is reinitialized using $\omega_{D,i}(k)$. One possible special case is that all sub-PLLs are unstable. In this case, all the sub-PLLs are reinitialized using $\bar{\omega}_{D,n}(k)$. The reinitialization operation leads to the sub-PLL under temporary open-loop control. The control signal is updated by the stable sub-PLL for $\omega_{D,i}(k)$ or the $\bar{\omega}_D$ estimator for $\bar{\omega}_{D,n}(k)$. Normally, the sub-PLL (or traditional PLL) is a closed-loop system. The feedback signal of the closed-loop PLL is the control signal that can be adjusted to the input signal. Therefore, when the input signal contains large error under scintillation conditions, the closed-loop sub-PLL may misestimate the control signal and become divergent. The open-loop PLL has a relatively better control signal under this case and thus keeps it in a stable state.

### 3.2. Output Fusion

The function of output fusion is to maintain the continuity of outputting accurate Doppler frequency measurements. It integrates $n$ redundant measurements into a single output. Considering $\varepsilon_{nn}(k) = v_n(k)$ as shown in (4), $\gamma_{nn}(k)$ gives an approximate magnitude of tracking error of sub-PLL $n$. If $\gamma_{nn}(k)$ approaches 0, it is very likely that sub-PLL $n$ contains large tracking error. On the contrary, if $\gamma_{nn}(k)$ approaches to 1, sub-PLL $n$ is considered accurate. Thus, the output fusion can be written as

$$\hat{\omega}^{\text{output}}(k) = \begin{cases} \hat{\omega}_{D,n}(k) & \gamma_{nn} = \max\{\gamma_{11}, \gamma_{22}, \ldots, \gamma_{NN}\} \\ \bar{\omega}_{D,1}(k) & \gamma_{11} = \gamma_{22} = \ldots = \gamma_{NN} = 0 \end{cases} \tag{8}$$

The output fusion shown in (8) aims at obtaining the $\hat{\omega}_{D,n}(k)$ from the sub-PLL $n$ that has the largest $\gamma_{nn}$ among all the sub-PLLs. Large $\gamma_{nn}$ implies that sub-PLL $n$ is in a stable state. Thus, this output fusion can provide a reliable Doppler frequency measurement. Apparently, the accuracy of $\hat{\omega}^{\text{output}}(k)$ depends on the accuracy of the selected $\hat{\omega}_{D,n}(k)$. If the selected $\hat{\omega}_{D,n}(k)$ has the highest accuracy among $\hat{\omega}_{D,1}(k), \ldots, \hat{\omega}_{D,N}(k)$, the Doppler frequency measurements obtained from the output fusion have the highest accuracy. However, it should be noted that the selection of $\hat{\omega}_{D,n}(k)$ is based on the value of $\gamma_{nn}$ which can largely represent the accuracy of $\hat{\omega}_{D,n}(k)$ but it is absolutely not a precise one. The most accurate $\hat{\omega}_{D,n}(k)$ is not necessarily corresponding to the largest $\gamma_{nn}$.

Similar to the tracking fusion, the asynchronous problem also exists in the output fusion since the period of $\hat{\omega}_{D,n}(k)$ outputted from each sub-PLL's tracking fusion is the same as each sub-PLL's integration time. Considering the integration time = 1, 5, 10, or 20 ms, we select the longest integration time among all the sub-PLLs as the common sampling period of all the output fusions. Only at the sampling moment does the output fusion integrates all the $\hat{\omega}_{D,n}(k)$ from all the sub-PLLs and outputs the $\hat{\omega}^{\text{output}}(k)$ according to (8). In each sub-PLL, the $\hat{\omega}_{D,n}(k)$ generated at other moments will not be considered.

## 4. Tests and Results

To evaluate the performance of the proposed multi-PLL tracking method, three sets of real-world GPS IF (intermediate frequency) data are collected using a system shown in Figure 4 and are analyzed. This system uses a GPS/Galileo L1 USB Front End with OCXO (Model ET09/C) developed by iP-solutions (http://www.ip-solutions.jp) to collect GPS radio frequency signals from an antenna. It outputs the digital IF signals via a USB cable to a computer. The sampling frequency is 16.3676 MHz, and IF is 4.1304 MHz.

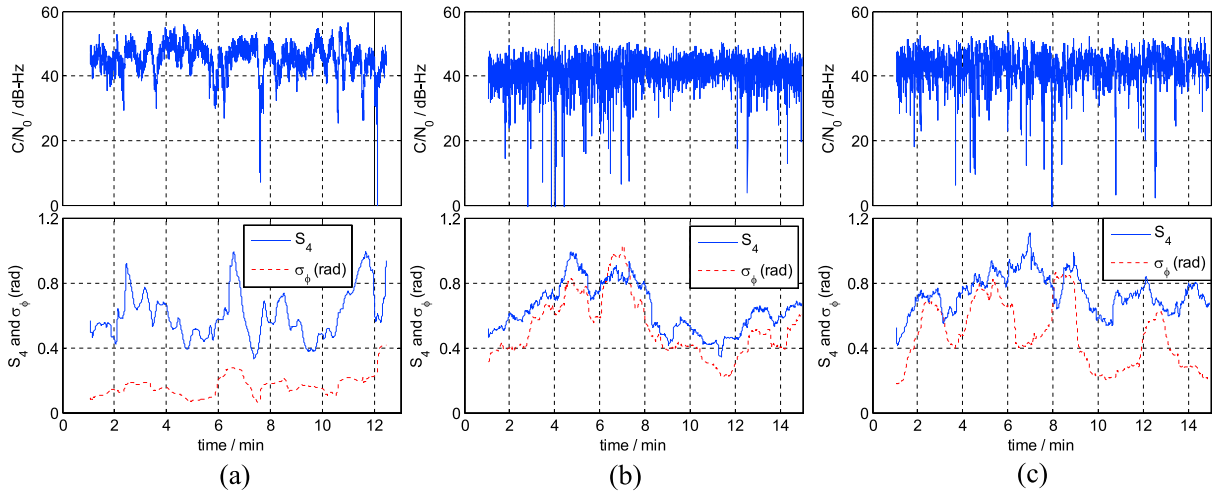


**Figure 5.** $C/N_0$, $S_4$, and $\sigma_\phi$ measured by second-order PLL with $B_n = 5$ Hz and $T_{coh} = 1$ ms for the three scintillation events: (a) event 1, (b) event 2, and (c) event 3.

The GPS IF data were collected in 2012, on the rooftop of the Lui Che Woo building at the Hong Kong Polytechnic University, Hong Kong. Among the data sets, three of them with strong scintillations were selected for this test. Scintillation event 1 was observed from PRN 5 during 21:13–21:26 LT (local time = UTC + 8 h) on 27 September 2012. Events 2 and 3 were observed from PRN 27 during 20:07–20:22 on 3 October 2012 and PRN 29 during 20:41–20:56 on 19 October 2012. The data of $C/N_0$ (estimated using variance summing method [*Sharawi et al.*, 2007]), $S_4$, and $\sigma_\phi$ are displayed in Figure 5. The first minute is used to calculate a converged solution of $S_4$ and $\sigma_\phi$; thus, it is not shown in Figure 5. In the three events, scintillations were at moderate or strong levels and led to $>20$ dB Hz $C/N_0$ degradation.

In the three scintillation events, the performance of the traditional single-PLL depends on its loop parameters, such as bandwidth and integration time. Table 1 shows the robustness of the traditional PLL with different loop parameters under the three events.

In Table 1, PLLs 4–6 with short integration time (1 ms) maintain signal tracking in all the three scintillation events. With long integration time (10 ms), PLLs 1–3 lose signal in one or more scintillation events. It is noticed that if the bandwidth is reduced to 3 Hz (integration time = 10 ms), PLL is able to track the scintillation-affected signals in all the three events. Narrow bandwidth is able to improve the robustness of tracking loop under scintillation condition but at the expense of poor dynamic performances. In the study, we focus on improving robustness of carrier tracking loop using traditional PLLs without complex loop parameters selection. Therefore, PLLs 1–3 are selected to form the multi-PLL though the performance of individual PLL is poor. It is anticipated that by adopting the algorithm developed in this paper, the multi-PLL is able to outperform the individual PLLs 1–3. To test the robustness of multi-PLL with respect to the loop parameters, another scheme by selecting PLLs 1, 2, and 6 is implemented. Within this scheme, PLL 6 has demonstrated strong robustness under scintillation conditions. PLL 1 has the lowest noise level under nonscintillation conditions due to its narrow bandwidth and long integration time.

**Table 1.** Tracking Results of Traditional Single-PLL[a]

| | Tracking Loop | | Events | | |
|---|---|---|---|---|---|
| | Integration Time (ms) | Bandwidth (Hz) | 1 | 2 | 3 |
| PLL 1 | 10 | 5 | x | x | y |
| PLL 2 | 10 | 10 | y | y | y |
| PLL 3 | 10 | 15 | y | x | y |
| PLL 4 | 1 | 5 | x | x | x |
| PLL 5 | 1 | 10 | x | x | x |
| PLL 6 | 1 | 15 | x | x | x |

[a]x: successful tracking; y: unsuccessful tracking.

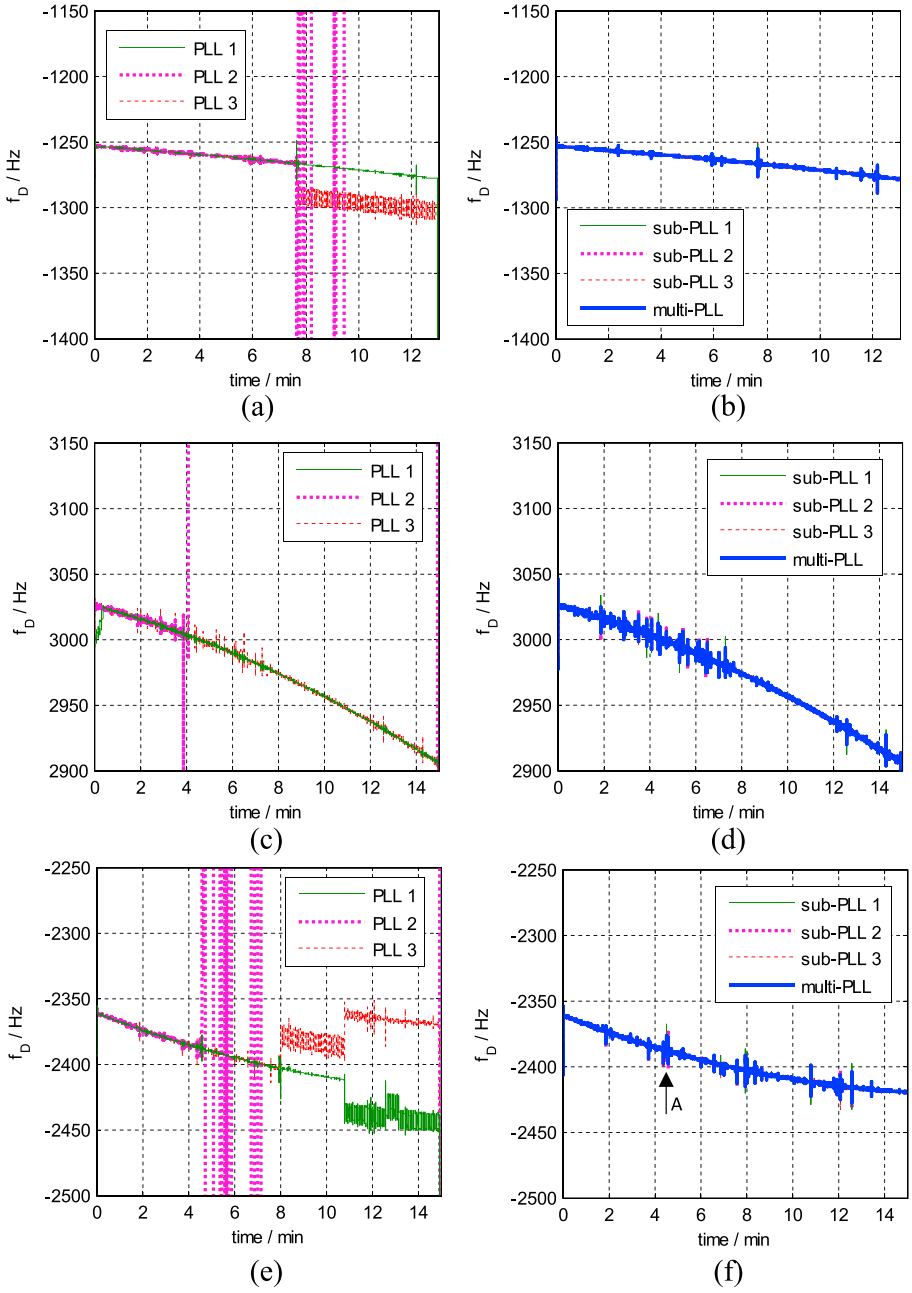


**Figure 6.** Estimated Doppler frequency from single-PLLs 1–3 and multi-PLL formed by sub-PLLs 1–3: (a) single-PLL in event 1, (b) multi-PLL in event 1, (c) single-PLL in event 2, (d) multi-PLL in event 2, (e) single-PLL in event 3, and (f) multi-PLL in event 3. The period of data marked with A in Figure 6f is shown in Figure 7.

In the following description, prefix "sub-" is used to denote a PLL used in the multi-PLL. For instance, sub-PLL 3 denotes the PLL with $B_n = 15$ Hz and $T_{coh} = 10$ ms working in the multi-PLL, but the PLL 3 denotes the traditional single-PLL with the same parameters: $B_n = 15$ Hz and $T_{coh} = 10$ ms.

Figure 6 shows the tracked Doppler frequency from traditional single-PLLs 1–3 and the multi-PLL formed by sub-PLLs 1–3 in the three scintillation events. As indicated in Figure 6 and Table 1, the single-PLL 2 is the most vulnerable one among the three single tracking loops because it loses signals in all the three scintillation events. PLL 1 fails in signal tracking in event 3, and PLL 3 loses signals in events 1 and 3. Combining PLLs 1–3, the multi-PLL significantly improves the performance of signal tracking. It helps its sub-PLLs overcome the scintillation effect and keeps signal tracking in all the three scintillation events.

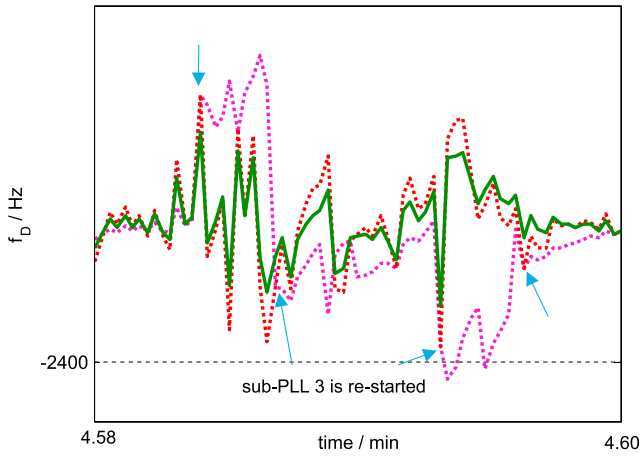


**Figure 7.** Doppler frequency variation of sub-PLLs during period A.

The improvement of multi-PLL on the robustness of carrier tracking loop is realized by using the tracking fusion. The tracking fusion first detects the state of each sub-PLL. If any sub-PLL is found unstable, it will be restarted. One example is displayed in Figure 7. Sub-PLL 3 is restarted 4 times, and each reinitialization value is obtained from sub-PLL 2 during period A (marked in Figure 6f). If all sub-PLLs are considered unstable, they would be restarted by using the output of $\omega_D^-$ estimator as the reinitialization value.

Figure 8 displays the tracked Doppler frequency from the traditional single-PLLs 1, 2, and 6 and multi-PLL in the three real-world scintillation events. In the multi-PLL, the three sub-PLLs succeed in keeping signal tracking in all the scintillation events and the multi-PLL can continuously output Doppler frequency measurements. This indicates that even with a selection of different sub-PLLs, the multi-PLL algorithm is still able to perform stably and has a robust carrier tracking loop under scintillation conditions.

In addition, Figure 8 shows that multi-PLL is able to overcome the shortcomings of single-PLLs. In Figure 8, single-PLLs are vulnerable or very noisy under scintillations. Single-PLLs 1 and 2 are vulnerable as discussed above. PLL 6 is able to track signals in all the three scintillation events, but its Doppler frequency measurements have much higher noises. For instance, standard deviation of the Doppler frequency error of PLL 6 is 1.22 Hz in event 1. The value is much larger than 0.33 Hz of PLL 1 and 0.25 Hz of PLL 2 (before PLL 2 unlock). Compared to single-PLL 6, the multi-PLL delivers more accurate measurements and the standard deviation of the Doppler frequency error is 0.40 Hz. In short, the multi-PLL significantly enhances the robustness though it can result in a slight degradation in the accuracy.

## 5. Conclusions

Loss of lock is a serious problem in GPS carrier tracking loop under ionospheric scintillation conditions. To reduce the occurrence frequency of loss of lock, a multi-PLL with two-stage fusion tracking algorithm is proposed in the paper. It is implemented by combining several parallel sub-PLLs via a two-stage fusion: tracking fusion and output fusion. Each sub-PLL has its own discriminator, loop filter, carrier NCO, and a tracking fusion. In the first stage, the tracking fusion is used to detect the state (stable or unstable) of each sub-PLL based on the degree of compatibility and generate a reliable feedback Doppler frequency. For stable sub-PLLs, their Doppler frequency measurements are directly used as feedback Doppler frequencies, while for unstable sub-PLLs the feedback Doppler frequency is taken from Doppler frequency measurements of other stable sub-PLLs. The output fusion, the second stage fusion of this algorithm, is used to output Doppler frequency measurements from sub-PLL with the largest degree of compatibility $\gamma_{nn}$.

The multi-PLL algorithm has been implemented and tested using two schemes of sub-PLL selection. One scheme selects PLL 1 ($B_n = 5$ Hz and $T_{coh} = 10$ ms), PLL 2 ($B_n = 10$ Hz and $T_{coh} = 10$ ms), and PLL 3 ($B_n = 15$ Hz and $T_{coh} = 10$ ms), and the other scheme selects PLL 1, PLL 2, and PLL 6 ($B_n = 15$ Hz and $T_{coh} = 1$ ms). The performance of multi-PLL has been evaluated using three sets of real-world GPS IF data recorded under moderate to strong ionospheric scintillations. The testing results show that the multi-PLL can significantly improve the reliability of carrier tracking loop and generate continuous Doppler frequency measurements. In the test, the multi-PLL is able to keep tracking in real-world scintillation cases ($S_4 = 0.26$–$1.1$ and $\sigma_\phi = 0.05$–$1.49$ rad). This improvement is because of the parallel architecture of multi-PLL algorithm. When one sub-PLL loses signal, its tracking fusion detects its unstable state and estimates an approximate Doppler frequency using the Doppler frequencies from other locked sub-PLLs as well as $\omega_D^-$ estimator. Then, the unstable sub-PLL is restarted using the approximate Doppler frequency, which helps the sub-PLL

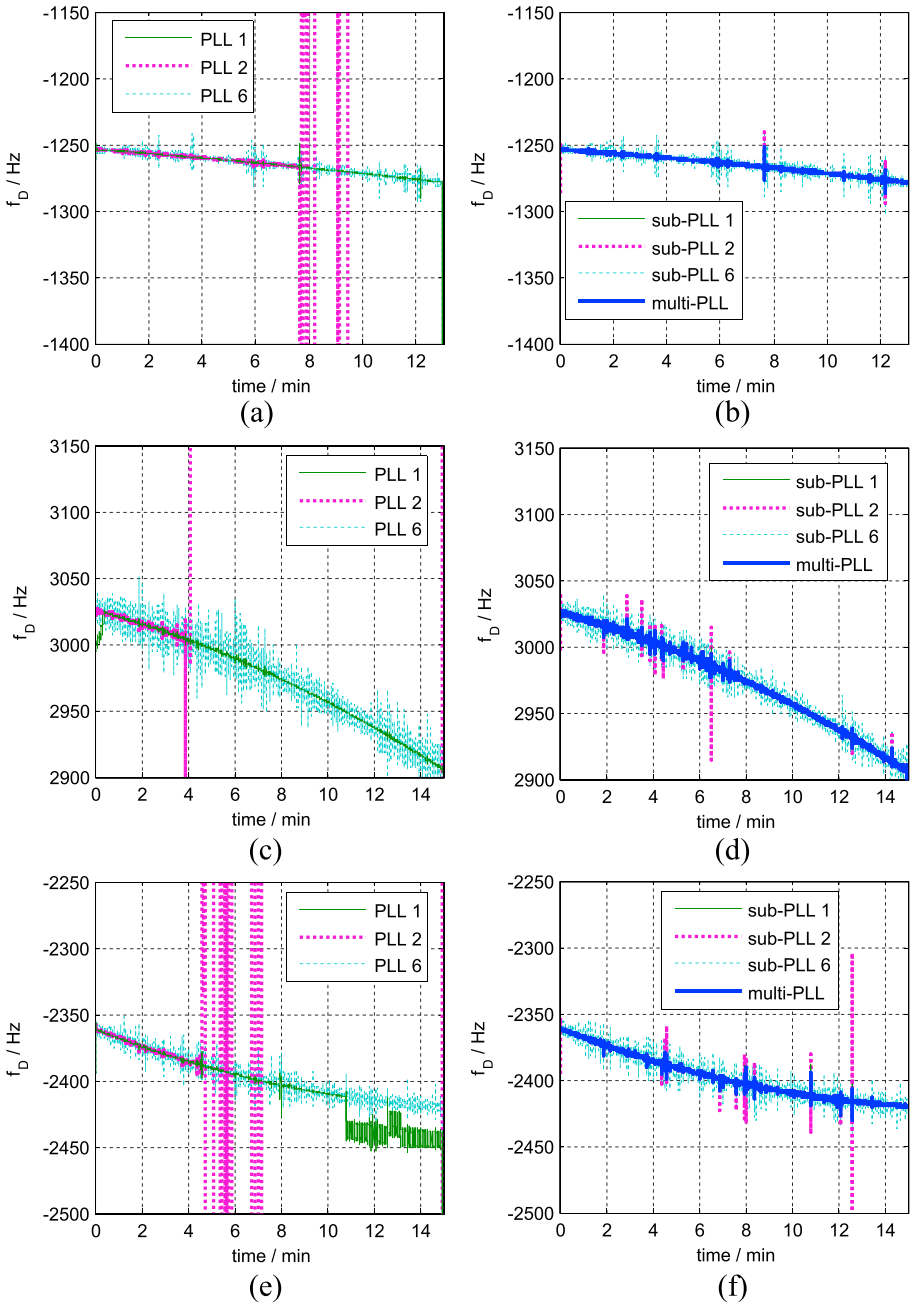


**Figure 8.** Estimated Doppler frequency from single-PLLs 1, 2, and 6 and multi-PLL formed by sub-PLLs 1, 2, and 6: (a) single-PLL in event 1, (b) multi-PLL in event 1, (c) single-PLL in event 2, (d) multi-PLL in event 2, (e) single-PLL in event 3, and (f) multi-PLL in event 3.

return to the stable state. In short, the multi-PLL algorithm has demonstrated its improved and robust performance over traditional single-PLL in maintaining reliable signal tracking even under strong ionospheric scintillation conditions.

This paper presents the preliminary study on the multi-PLL. We implemented it using three traditional PLLs. The multi-PLL improves robustness of carrier tracking loop under scintillation at the cost of a slight degradation of measurement accuracy. In the future studies, this multi-PLL method will be tested in more challenging conditions such as reduced $C/N_0$ (using low antenna gain), deep amplitude fading, and phase scintillations. To maintain similar level of performance, it is suggested that using advanced PLLs in each

sub-PLL (i.e., the adaptive PLL and Kalman filter-based PLL), increasing the number of sub-PLLs and improving the output fusion accuracy can be considered in further work of multi-PLL research.

## Appendix A: Preprocessing Before the Tracking Fusion

In the implementation of multi-PLL algorithm, it is probable that each sub-PLL has a different integration time. To handle this, a preprocessing is needed to synchronize the outputs of different sub-PLLs. Without loss of generality, we choose sub-PLL $n$ as an example. The integration time of sub-PLL $n$ is $T_{\mathrm{coh},n}$, and its noise is modeled as zero mean Gaussian distribution with variance $\sigma_{v,n}^2\left(v_n(k_n) \sim N\left(0, \sigma_{v,n}^2\right)\right)$. In the multi-PLL, the output of sub-PLL $n$ should be synchronized with those of other sub-PLLs. Assuming that one of the sub-PLLs, for example, sub-PLL $m$ has an integration time $T_{\mathrm{coh},m}$ and its noise is modeled as $v_m(k_m) \sim N\left(0, \sigma_{v,m}^2\right)$. If $T_{\mathrm{coh},m} = T_{\mathrm{coh},n}$, sub-PLLs $m$ and $n$ are synchronous and the preprocessing is not needed. Otherwise, when $T_{\mathrm{coh},m} \neq T_{\mathrm{coh},n}$, preprocessing is needed to synchronize the output of sub-PLL $m$ with that of sub-PLL $n$. This preprocessing is implemented according to the cases of $T_{\mathrm{coh},m} > T_{\mathrm{coh},n}$ or $T_{\mathrm{coh},m} < T_{\mathrm{coh},n}$.

If $T_{\mathrm{coh},m} > T_{\mathrm{coh},n}$, sub-PLL $m$ outputs its Doppler frequency slower than sub-PLL $n$. To synchronize them, it is required to upsample the output of sub-PLL $m$. Thus, the preprocessing is implemented by

$$\omega_{D,m}^{s}(k_n) = \omega_{D,m}(\lfloor k_n/M \rfloor) \tag{A1}$$

where $k_n = 0, 1, 2, \ldots$; $\omega_{D,m}^{s}(k_n)$ represents $\omega_{D,m}$ with a new sampling rate of $1/T_{\mathrm{coh},n}$ Hz, which is the same as sub-PLL $n$; $\omega_{D,m}(\lfloor k_n/M \rfloor)$ means that $\omega_{D,m}(k_m)$ of original sampling rate of $1/T_{\mathrm{coh},m}$ Hz is resampled by time sequence $k_n$; $M = T_{\mathrm{coh},m}/T_{\mathrm{coh},n}$ is a positive integer. To ensure $M$ to be an integer for any two sub-PLLs in a multi-PLL, the integration time of a sub-PLL is recommended as 1, 5, 10, or 20 ms. These values are also commonly used in traditional PLLs. Equation (A1) shows that the function of the preprocessing is to upsample the output of sub-PLL $m$ by $M$ times, from $1/T_{\mathrm{coh},m}$ Hz to $1/T_{\mathrm{coh},n}$ Hz. This process does not affect the characteristics of the noise of sub-PLL $m$. After preprocessing, $v_m(k_n)$ follows $N\left(0, \sigma_m^2\right)\left(\sigma_m^2 = \sigma_{v,m}^2\right)$.

If $T_{\mathrm{coh},m} < T_{\mathrm{coh},n}$, sub-PLL $m$ outputs its Doppler frequency faster than sub-PLL $n$. The preprocessing is required to downsample the output of sub-PLL $m$, which is implemented as

$$\omega_{D,m}^{s}(k_n) = \frac{1}{M}\sum_{i=0}^{M-1} \omega_{D,m}(Mk_n + i) \tag{A2}$$

The ratio $M = T_{\mathrm{coh},n}/T_{\mathrm{coh},m}$ should also be an integer. The preprocessing downsamples the output of sub-PLL $m$ by $M$ times. In this case, the model of noise $v_m(k_n)$ is $N\left(0, \sigma_m^2\right)\left(\sigma_m^2 = \sigma_{v,m}^2/M\right)$.

**Acknowledgments**

This work is supported by the National Natural Science Foundation of China (NSFC project 41274039), the Construction Industry Institute Hong Kong/PolyU Innovation Fund Governing Council (project 5-ZJD5), and the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) (project PolyU 5203/13E, B-Q37X). Zhizhao Liu also thanks the support of the Program of Introducing Talents of Discipline to Universities (Wuhan University, GNSS Research Center), China. Three anonymous reviewers were thanked for their comments and suggestions for improving the quality of this paper. The test data sets used in this work can be freely obtained upon request by contacting the authors.

## References

Chiou, T., D. Gebre-Egziabher, T. Walter, and P. Enge (2007), Model analysis on the performance for an inertial aided FLL-assisted-PLL carrier-tracking loop in the presence of ionospheric scintillation, in *Proceedings of ION-NTM-2007*, pp. 1276–1295, Institute of Navigation, Savannah, Ga., 22–24 Jan.

Chiou, T., J. Seo, T. Walter, and P. Enge (2008), Performance of a Doppler-aided GPS navigation system for aviation applications under ionospheric scintillation, in *Proceedings of ION-GNSS-2008*, pp. 1139–1147, Institute of Navigation, Savannah, Ga.

Crane, R. (1977), Ionospheric scintillation, *Proc. IEEE*, *52*(August), 183–208, 16–19 Sept.

De Oliveira Moraes, A., F. Silveira Rodrigues, W. J. Perrella, and E. R. Paula (2011), Analysis of the characteristics of low-latitude GPS amplitude scintillation measured during solar maximum conditions and implications for receiver performance, *Surv. Geophys.*, *33*(5), 1107–1131, doi:10.1007/s10712-011-9161-z.

Forte, B. (2012), Analysis of strong ionospheric scintillation events measured by means of GPS signals at low latitudes during disturbed conditions, *Radio Sci.*, *47*, RS4009, doi:10.1029/2011RS004789.

Ganguly, S., A. Jovancevic, A. Brown, M. Kirchner, S. Zigic, T. Beach, and K. Groves (2004), Ionospheric scintillation monitoring and mitigation using a software GPS receiver, *Radio Sci.*, *39*, RS1S21, doi:10.1029/2002RS002812.

Gao, Y., X. Liao, and Z. Liu (2002), Ionosphere modelling using carrier smoothed ionosphere observations from a regional GPS network, *Geomatica*, *56*(2), 97–106.

Humphreys, T. (2005), GPS carrier tracking loop performance in the presence of ionospheric scintillations, in *Proceedings of ION-GNSS-2005, September 13–16*, pp. 156–167, Institute of Navigation, Long Beach, Calif.

Humphreys, T. E., M. L. Psiaki, and P. M. Kintner (2010), Modeling the effects of ionospheric scintillation on GPS carrier phase tracking, *IEEE Trans. Aerosp. Electron. Syst.*, *46*(4), 1624–1637, doi:10.1109/TAES.2010.5595583.

Leick, A. (2004), *GPS Satellite Surveying*, 3rd ed., John Wiley, Hoboken, N. J.

Luo, W., Z. Liu, and M. Li (2014), A preliminary evaluation of the performance of multiple ionospheric models in low- and mid-latitude regions of China in 2010–2011, *GPS Solutions*, *18*(2), 297–308, doi:10.1007/s10291-013-0330-z.

Morrissey, T. N., K. W. Shallberg, A. J. Van Dierendonck, and M. J. Nicholson (2004), GPS receiver performance characterization under realistic ionospheric phase scintillation environments, *Radio Sci.*, *39*, RS1S20, doi:10.1029/2002RS002838.

Sharawi, M. S., D. M. Akos, and D. N. Aloi (2007), GPS $C/N_0$ estimation in the presence of interference and limited quantization levels, *IEEE Trans. Aerosp. Electron. Syst.*, *43*(1), 227–238.

Skone, S., G. Lachapelle, and D. Yao (2005), Investigating the impact of ionospheric scintillation using a GPS software receiver, in *Proceedings of ION-GNSS-2005*, pp. 1126–1137, Institute of Navigation, Long Beach, Calif., 13–16 Sept.

Van Dierendonck, A. J., J. Klobuchar, and Q. Hua (1993), Ionospheric scintillation monitoring using commercial single frequency C/A code receivers, in *Proceedings of ION-GPS-1993*, pp. 1333–1342, Institute of Navigation, Salt Lake City, Utah, 22–24 Sept.

Won, J., and B. Eissfeller (2013), A tuning method based on signal-to-noise power ratio for adaptive PLL and its relationship with equivalent noise bandwidth, *IEEE Commun. Lett.*, *17*(2), 393–396, doi:10.1109/LCOMM.2013.01113.122503.

Won, J., T. Pany, and B. Eissfeller (2012), Characteristics of Kalman filters for GNSS signal tracking loop, *IEEE Trans. Aerosp. Electron. Syst.*, *48*(4), 3671–3681, doi:10.1109/TAES.2012.6324756.

Xu, G. (2007), *GPS, Theory, Algorithms and Applications*, 2nd ed., Springer, Berlin.

Xu, R., Z. Liu, and W. Chen (2014), Improved FLL-assisted PLL with in-phase pre-filtering to mitigate amplitude scintillation effects, *GPS Solutions*, *19*, 263–276, doi:10.1007/s10291-014-0385-5.

Yeh, W. C., and T. J. Hsieh (2011), Solving reliability redundancy allocation problems using an artificial bee colony algorithm, *Comput. Oper. Res.*, *38*(11), 1465–1473, doi:10.1016/j.cor.2010.10.028.

Zhang, L., and Y. Morton (2009), Tracking GPS signals under ionosphere scintillation conditions, in *Proceedings of ION-GNSS-2009, September 22–25*, pp. 227–234, Institute of Navigation, Savannah, Ga., 22–25 Sept.

Zhang, L., Y. Morton, F. van Graas, and T. Beach (2010), Characterization of GNSS signal parameters under ionosphere scintillation conditions using software-based tracking algorithms, in *IEEE/ION-PLANS-2010*, pp. 264–275, IEEE, Indian Wells, Calif., doi:10.1109/PLANS.2010.5507209, 4–6 May.