

A Fast Acquisition Algorithm Based on Division of GNSS Signals

Qingxi Zeng¹, Wenqi Qiu¹, Pengna Zhang¹, Xuefen Zhu² and Ling Pei³

¹*(Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Jiangsu, P. R. China)*

²*(School of Instrument Science and Engineering, Southeast University, Jiangsu, P.R. China)*

³*(Shanghai Key Laboratory of Navigation and Location Based service, Shanghai Jiao tong University, Shanghai, P. R. China)*

(E-mail: jslyzqx@163.com)

The acquisition of signals is a precondition for tracking and solution calculation in software-based Global Navigation Satellite System (GNSS) receivers. The Parallel Code phase Acquisition (PCA) algorithm can simultaneously obtain the correlation results at every sampling point. However, if the number of sampling points that needs processing is large, this method will lead to a heavy computational load. Thus, we improve the process of the PCA algorithm and propose a novel algorithm that divides the signals into K (K is a constant) parts to achieve correlation and obtains the correlation results with a fusion algorithm. This algorithm can simultaneously obtain the correlation results for sampling points at an interval of K points. If the K value is selected appropriately, the computational load can be decreased by about 50%. Also, the Receiver Operating Characteristic (ROC) curves show that under a certain probability of false alarm, the detection probability of the proposed algorithms is 5% lower than that of the PCA algorithm. Therefore, the proposed algorithm can speed up the acquisition process with a slight decrease in detection probability.

KEY WORDS

1. Software receiver.
2. BeiDou.
3. Fast Fourier Transform (FFT).

Submitted: 2 September 2016. Accepted: 3 December 2017. First published online: 2 February 2018.

1. INTRODUCTION. The BeiDou Navigation Satellite System, which is a Chinese global satellite navigation system, has gradually been brought online in the last few years. The China Satellite Navigation Office (2013) has released the Interface Control Document (ICD) for open service signals B1 and B2. This indicates that BeiDou can provide users with a dual-frequency high precision positioning service. A satellite navigation receiver can receive the signals broadcast by the satellites in view and process them to provide location and navigation information. The key part is baseband signal processing, which is based

on the acquisition algorithm. The operating time of the acquisition algorithm will directly affect the Time To First Fix (TTFF) of the receiver (Jan and Lin, 2009). C_{B2I} is the same as the Coarse/Acquisition (C/A) code of the Global Positioning System (GPS) in sequence generation, which is generated by two series of Gold code sequences. The difference is that the Gold code in C_{B2I} is generated through 11 shift registers, while the Gold code in C/A code is generated from ten shift registers. The code rate of BeiDou C_{B2I} code is twice that of GPS C/A code. In order to ensure acquisition accuracy, the sampling rate of the B2 signal needs to be increased accordingly. The number of sampling points to be processed by the acquisition algorithm will also increase. Using the traditional algorithm will extend the acquisition time. Therefore, it is necessary to study the fast acquisition algorithm (Simone et al., 2011) to reduce the receiver delay (Zhang et al., 2016; Jin et al., 2015).

When the software receiver demodulates and spreads received signals, there are three important parameters: the number of satellites in view, the Doppler frequency offset and the code phase shift. The process of signal acquisition can be seen as a three-dimensional search for the above three parameters (Tsui, 2005). The Serial Search Acquisition (SSA) algorithm mainly takes advantage of the correlation property of the ranging code (that is the C/A code or C_{B2I} code) and scans all the cells in the search space (Polydoros and Weber, 1984). To speed up the acquisition process, Van Nee and Coenen (1991) applied a Fast Fourier Transform (FFT) to the correlation of the ranging code and proposed the Parallel Code phase Acquisition (PCA) algorithm. This converted correlation in the time-domain to a multiplication in the frequency-domain and obtained correlation results at every sampling point at one time. Thus, only a two-dimensional search for the received signal was needed. In this way, the computational load was significantly reduced. However, the computational load is still large when the satellite signal is long or the sampling frequency is high, so many researchers have studied acquisition algorithms based on PCA algorithms. For instance, Starzyk and Zhu (2001) combined an Averaging Correlations (AC) method with the PCA algorithm and proposed the AC-PCA algorithm. This divided the 5,000-point FFT to six 1,023-point FFTs in the correlation operation. Fantino et al. (2008) analysed the performance of this algorithm thoroughly, and this work shows that the process speed can be improved without loss of energy. Chun (2005) applied segmented FFT to the acquisition of L2 signals. In this algorithm, L2 signals were divided into small segments and processed after arriving at the receiver. In this way, the latency, memory and computation requirements per segment were reduced.

Lin et al. (1999) proposed the Double-Block Zero-Padding (DBZP) method based on a delay-and-multiply approach. This algorithm was used to capture the P(Y)-Code directly. It broke down the long data into M subsets and performed local correlation in the frequency domain, where M was the number of Doppler bins. In this way, the length of the FFT block was decreased and the calculation time could be reduced. Jan and Lin (2009) proposed a multi-C/A code acquisition method. It replaced the single local C/A code in the PCA algorithm to a sum of two or more C/A codes of different satellites. This can search for several satellites at one time and speed up the acquisition process. Leclère et al. (2012) combined the Fast Finite Impulse Response (FIR) Algorithms (FFAs) and PCA algorithm into a new algorithm. This algorithm decomposed the initial circular correlation into several smaller circular correlations and reduced the length of the FFT blocks. Patel and Shukla (2011) exploited the symmetry of the power spectrum of the C/A code to reduce the number of points to perform correlation and FFT. The reduction in FFT/Inverse FFT (IFFT) points can decrease the computation time and hardware complexity. The above-mentioned

algorithms all improved the PCA algorithm and reduced the computational load or resource consumption, but at the cost of increasing algorithm complexity or sometimes decreasing acquisition accuracy.

In this paper, we propose a new acquisition algorithm. It divides the signal into K parts to conduct FFT, and obtains the correlation results in the frequency domain with a fusion algorithm. This algorithm can shift K sampling points in one correlation operation. What is more, it can also be applied to capture any satellite signals modulated in Binary Phase Shift Keying (BPSK) and Quadrature Phase Shift Keying (QPSK) modes with a lower computational load. In this paper, this algorithm will be denoted as the K -PCA algorithm.

The rest of the paper is organised as follows. In Section 2, we briefly introduce a model of the B2 signal. A quick review of the SSA and PCA algorithms is presented in Section 3. Then the principle of the K -PCA algorithm is presented in Section 4. In Section 5, we evaluate the computational complexity of the K -PCA algorithm and the PCA algorithm. The method to choose the optimal K value is also described in this section. Section 6 shows the algorithm operation simulation results of the K -PCA algorithm and conclusions are drawn in Section 7.

2. MODEL OF THE B2 SIGNAL. The China Satellite Navigation Office (2013) has released the ICD for open service signal B2, describing its structure, basic parameters, ranging code features and navigation data format. Unlike the BPSK modulation of the GPS signal, BeiDou adopts the QPSK modulation mode (China Satellite Navigation Office, 2013). Compared with BPSK, the QPSK transmits In-phase/Quadrature-phase (I/Q) signals simultaneously and utilises the frequency band more efficiently. The B2 signal consists of the I branch (open) and the Q branch (authorisation), and can be expressed as:

$$S_{B2}^j(t) = A_{B2I} C_{B2I}^j(t) D_{B2I}^j(t) \cos(2\pi f_{B2} t + \varphi_{B2I}^j) + A_{B2Q} C_{B2Q}^j(t) D_{B2Q}^j(t) \sin(2\pi f_{B2} t + \varphi_{B2Q}^j) \quad (1)$$

In this equation, superior j represents the serial number of the satellite. A_{B2I} and A_{B2Q} indicate the signal intensity of the I branch and the Q branch. C_{B2I} and C_{B2Q} are ranging codes modulated in the I/Q branches. D_{B2I} and D_{B2Q} are navigation data. φ_{B2I} and φ_{B2Q} are carrier phases of the I/Q branches respectively. The carrier frequency f_{B2} is 1207.14 MHz. This paper only studies the I branch signal, as the Q branch signal needs authorisation and is not open to civil use. The C_{B2I} code is a Gold sequence composed of 2,046 chips, and the code rate is 2.046 Mcps.

The satellite signal is obtained by modulating the C_{B2I} code and navigation data with the carrier in the QPSK mode. Unlike the traditional QPSK mode, the B2 signal digital data (C_{B2I} code and navigation data) do not go through a multiplexer but multiply with the carrier of the I (or Q) branch directly (Xie et al., 2016). The modulation process is illustrated in Figure 1.

The code rate of C_{B2I} is twice that of the GPS C/A code. To ensure acquisition accuracy, the signal sampling frequency needs to be improved. Thus, the number of sampling points that need to be processed will increase too, which will affect the real-time quality of the receiver. The heavy computational load correspondingly will make the software receiver much more difficult to implement on an embedded platform with limited resources (Jin et al., 2008).

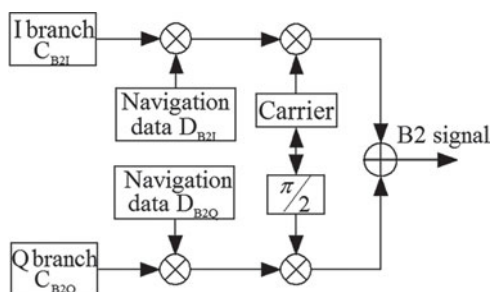


Figure 1. Modulation mode of B2 signal.

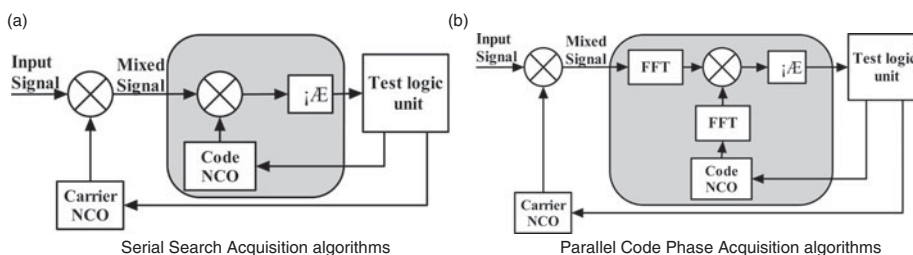


Figure 2. Principle diagrams of acquisition algorithms.

3. PRINCIPLES OF CONVENTIONAL ACQUISITION ALGORITHMS. When the signal is transmitted from the satellite to the GNSS receiver, its carrier frequency and code phase will change due to the Doppler effect and the unknown distance. Also, it will be submerged in noise. Thus, it is necessary for the GNSS receiver to raise the signal from the noise by acquisition blocks. The acquisition algorithm mainly takes advantage of the correlation property of the ranging code. The process of the acquisition algorithm is illustrated in Figure 2(a). Once the local carrier and Pseudorandom Number (PRN) code are synchronised with the received signal, there is a peak and the signal can be captured (Akopian, 2005).

The PCA algorithm converted the correlation in the time-domain to multiplication in the frequency-domain by using FFT, while the correlation result at every sampling point can be obtained simultaneously by applying IFFT to the product. The process is shown in Figure 2(b). For each satellite, this algorithm only requires searching for the Doppler frequency offset, so the computational load can be significantly reduced. Though the PCA approach is computationally efficient, it must compute the correlation results at every sampling point due to the FFT and IFFT. Thus, its structure is less flexible than the SSA algorithm illustrated in Figure 2(a), which can shift several sampling points in one correlation operation. Provided that the signal sampling frequency is relatively high or the number of signal sampling points that need to be processed is large (e.g. weak signals and long period signals), this algorithm will need large FFT processing blocks to process the data, which will increase the load on the receiver (Chun et al., 2006). Therefore, if we can combine the advantages of the SSA and PCA algorithms, the length of FFT blocks and computational load can be significantly decreased. Based on this idea, we improve the structure of the PCA - more precisely the structure of FFT blocks in the correlation

operation, and propose the following algorithm. It shifts K sampling points in every correlation operation, and the K value can be arbitrarily selected according to the situation; the algorithm is thus called the K -PCA algorithm.

4. THE CONCEPT OF K -PCA ALGORITHM.

4.1. *The principle of the 3-PCA algorithm (take $K = 3$ as an example).* The correlation (Wang et al., 2013) between mixed signal $x(n)$ and local C_{B2I} code $h(n)$ is shown in Equation (2).

$$y(n) = \sum_{m=0}^{N-1} x(m)h(m - n) \tag{2}$$

where $x(n)$, $h(n)$ and $y(n)$ are finite sequences whose lengths are N (N is the number of sampling points in one period of the signal). $x(n)$ means mixed signal $x(n)$, $h(n)$ means local C_{B2I} , and the period of $h(n)$ is 1 ms.

We then transform the correlation in the time domain to multiplication in the frequency domain by applying FFT to $y(n)$. The transformed result can be expressed as Equation (3), where $W_N^k = e^{-j2\pi kn/N}$ represents the twiddle factor, and $H^*(k)$ is the complex conjugate of $H(k)$.

$$\begin{aligned} Y(k) &= \sum_{n=0}^{N-1} y(n)W_N^{nk} = \sum_{n=0}^{N-1} \left(\sum_{m=0}^{N-1} x(m)h(m - n) \right) W_N^{nk} = \sum_{n=0}^{N-1} \left(\sum_{m=0}^{N-1} x(m)h(m - n) \right) W_N^{nk} W_N^{-(m-n)k} \\ &= \underbrace{\sum_{m=0}^{N-1} x(m)W_N^{mk}}_{X(k)} \underbrace{\sum_{n=0}^{N-1} h(m - n)W_N^{-(m-n)k}}_{H^*(k)} = X(k)H^*(k) \end{aligned} \tag{3}$$

Meanwhile, $X(k)$ can be represented as follows:

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n)W_N^{nk} = \sum_{\substack{n=0 \\ \text{mod}(n,3)=0}}^{N-1} x(n)W_N^{nk} + \sum_{\substack{n=0 \\ \text{mod}(n,3)=1}}^{N-1} x(n)W_N^{nk} + \sum_{\substack{n=0 \\ \text{mod}(n,3)=2}}^{N-1} x(n)W_N^{nk} \\ &= \sum_{r=0}^{N/3-1} x(3r)W_N^{3rk} + \sum_{r=0}^{N/3-1} x(3r + 1)W_N^{(3r+1)k} + \sum_{r=0}^{N/3-1} x(3r + 2)W_N^{(3r+2)k} \\ &= \underbrace{\sum_{r=0}^{N/3-1} x_0(r)W_{N/3}^{rk}}_{X_0(k)} + W_N^k \underbrace{\sum_{r=0}^{N/3-1} x_1(r)W_{N/3}^{rk}}_{X_1(k)} + W_N^{2k} \underbrace{\sum_{r=0}^{N/3-1} x_2(r)W_{N/3}^{rk}}_{X_2(k)} \\ &= X_0(k) + W_N^k X_1(k) + W_N^{2k} X_2(k) \end{aligned} \tag{4}$$

where

$$\begin{cases} x_0(r) = x(3r) \\ x_1(r) = x(3r + 1) \\ x_2(r) = x(3r + 2) \end{cases} \quad r \in [1, 2, \dots, N/3 - 1] \tag{5}$$

$$X_i(k) = FFT[x_j(r)] \quad i \in [0, 2]$$

Similarly,

$$H^*(k) = H_0^*(k) + W_N^{-k}H_1^*(k) + W_N^{-2k}H_2^*(k); \quad Y(k) = Y_0(k) + W_N^kY_1(k) + W_N^{2k}Y_2(k) \tag{6}$$

where

$$h_i(r) = h(3r + i), H_i^*(k) = [FFT[h_i(r)]]^* \quad i \in [0, 2], \quad r \in [1, 2, \dots, N/3 - 1] \tag{7}$$

According to Equations (3), (4) and (6), we can obtain:

$$\begin{aligned} Y(k) &= [X_0(k) + W_N^kX_1(k) + W_N^{2k}X_2(k)] \times [H_0^*(k) + W_N^{-k}H_1^*(k) + W_N^{-2k}H_2^*(k)] \\ &= \underbrace{X_0(k)H_0^*(k) + X_1(k)H_1^*(k) + X_2(k)H_2^*(k)}_{Y_0(k)} + \underbrace{W_N^{-2k}X_0(k)H_2^*(k) + W_N^kX_1(k)H_0^*(k) + W_N^k}_{W_N^kY_1(k)} \\ &\quad + \underbrace{W_N^{-k}X_0(k)H_1^*(k) + W_N^{-k}X_1(k)H_2^*(k) + W_N^{2k}X_2(k)H_0^*(k)}_{W_N^{2k}Y_2(k)} \end{aligned} \tag{8}$$

$Y_i(k)(i \in [0, 2])$ can be represented as:

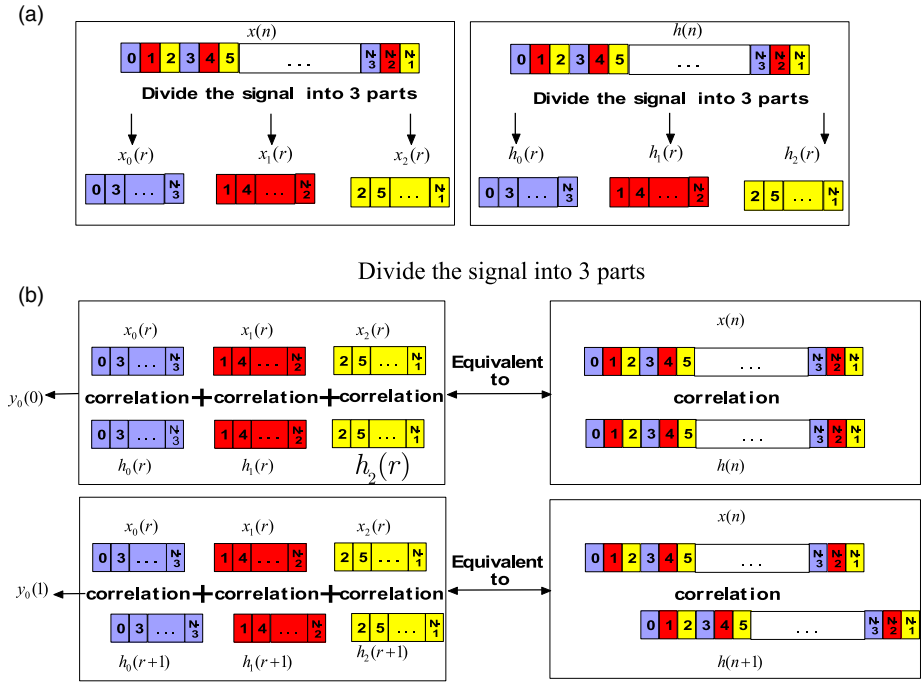
$$\begin{aligned} Y_0(k) &= H_0^*(k)X_0(k) + H_1^*(k)X_1(k) + H_2^*(k)X_2(k) \\ Y_1(k) &= H_0^*(k)X_1(k) + H_1^*(k)X_2(k) + W_N^{-3k}H_2^*(k)X_0(k) \\ Y_2(k) &= H_0^*(k)X_2(k) + W_N^{-3k}H_1^*(k)X_0(k) + W_N^{-3k}H_2^*(k)X_1(k) \end{aligned} \tag{9}$$

The result in the time domain can be derived through IFFT after $Y_i(k)(i \in [0, 2])$ is computed, and can be represented as:

$$y_i(n) = IFFT(Y_i(k)) \quad i \in [0, 2] \tag{10}$$

The final correlation result in the time domain can be obtained by combining $y_i(n)(i \in [0, 2])$.

Next, we will take $y_0(n)$ as an example to analyse the correlation process of the signal. The algorithm divides the mixed signal and local C_{B2I} code into three parts (Jin et al., 2013), which is shown in Figure 3(a), and the FFT operation is applied separately to each part. From Equation (9), we can see that, in the frequency domain, $Y_0(k)$ is a sum of three parts— $X_0(k) \times H_0^*(k)$, $X_1(k) \times H_1^*(k)$ and $X_2(k) \times H_2^*(k)$. In the time domain, $y_0(n)$ is a sum of three parts— $x_0(r) * h_0(r)$, $x_1(r) * h_1(r)$ and $x_2(r) * h_2(r)$ where $*$ means convolution. The computing process of $y_0(0)$ and $y_0(1)$ is shown in Figure 3(b). From Figure 3(a) we can see that when $h_i(r)(i \in [0, 2])$ shifts by one bit to the right, the $h(n)$ shifts by three bits



Computing process of $y_0(0)$ and $y_0(1)$

Figure 3. Computing process of $y_0(n)$.

to the right compared to the original sequence. So, in this algorithm, $y_0(n)$ is still the cross-correlation result between $x(n)$ and $h(n)$. The difference is that $h(n)$ shifts by three bits in every cross-correlation operation.

The computing process of $y_1(n)$ and $y_2(n)$ is similar to that of $y_0(n)$, but in Equation (9), $H_i^*(k)$ is multiplied by the twiddle factor $W_{N/3}^{-k}$. In light of the circumference-shifting characteristic of an FFT (Cheng, 2012), we can see that this operation is equal to shifting $h_i(r)$ to the right by one bit first, and then applying the FFT and the conjugate operation to it. The computing process of $y_i(n)$ is shown in Figure 4. The final correlation result $y'(n)$ can be obtained by combining $y_i(n)(i \in [0, 2])$ together, as shown in Equation (11).

$$y'(3n) = y_0(n); y'(3n + 1) = y_1(n); y'(3n + 2) = y_2(n) \quad n \in [0, N/3 - 1] \quad (11)$$

If the difference between local carrier frequency and received signal frequency is bigger than the search step (for example, 1 kHz or 500 Hz), there will be no peak in the correlation results between the mixed signal and local C_{B2I} code. Similarly, there will be no value exceeding the threshold in $y_0(n)$, $y_1(n)$ and $y_2(n)$. If the local carrier frequency is the same as the received signal frequency, there will be peaks in $y(n)$. Also, the correlation results in the sampling points near the code phase will exceed the threshold. The peaks in $y_0(n)$, $y_1(n)$ and $y_2(n)$ will also exceed the threshold. Hence, it can be judged whether the frequency of the local carrier is the same as the frequency of the received signal by the peak value of $y_0(n)$. When a correlation peak in $y_0(n)$ occurs that exceeds the threshold, it indicates that the

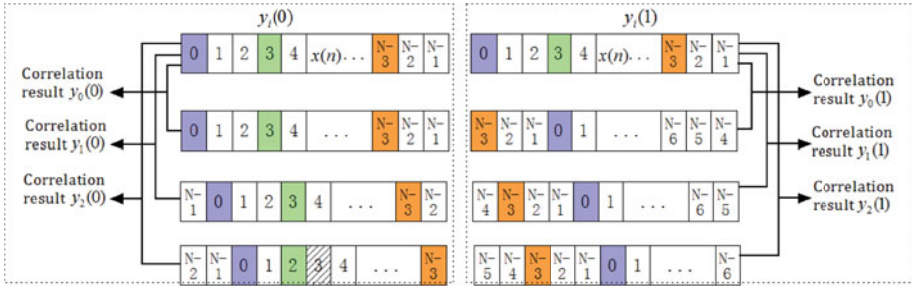


Figure 4. Computing process of $y_i(n)$.

search for frequency offset is accomplished. Otherwise, the frequency of the local carrier should be changed and the acquisition steps need to be repeated until a correlation peak appears. Afterwards, the value of $y_1(n)$ and $y_2(n)$ should be calculated. We then compare the peak values of $y_0(n)$, $y_1(n)$ and $y_2(n)$. If the biggest peak value is located at the m th points in $y_i(n)$ (m is a constant), the final code phase is located at the $(3m + i)$ th sampling point.

The acquisition process of the 3-PCA algorithm is demonstrated in Figure 5 and as follows.

Step 1: First, strip off the carrier from the received signal by the process shown in Equation (12), where S_{IF} represents the received signal and $L(n)$ represents the local carrier. Next, divide the mixed signal and local C_{B2I} code into three parts by the process shown in Equation (13).

$$x(n) = S_{IF}(n)L(n) \tag{12}$$

$$\begin{cases} x_0(r) = x(3 \times r) \\ x_1(r) = x(3 \times r + 1) \\ x_2(r) = x(3 \times r + 2) \end{cases} \quad \begin{cases} h_0(r) = h(3 \times r) \\ h_1(r) = h(3 \times r + 1) \\ h_2(r) = h(3 \times r + 2) \end{cases} \quad r \in \left[0, \frac{N}{3} - 1\right] \tag{13}$$

Step 2: Apply the FFT to $x_i(r)$, $h_i(r)$, $i \in \{0, 1, 2\}$, then take the conjugate value of $H_i(k)$. The procedures are as follows:

$$\begin{aligned} X_i(k) &= FFT(x_i(r)) \\ H_i(k) &= FFT(h_i(r)) \\ H_i^*(k) &= conj(H_i(k)) \quad i \in \{0, 1, 2\}, \quad r \in [1, 2, \dots, N/3 - 1] \end{aligned} \tag{14}$$

Step 3: Obtain $Y_0(k)$ according to the fusion algorithm shown in Equation (15).

$$Y_0(k) = X_0(k)H_0^*(k) + X_1(k)H_1^*(k) + X_2(k)H_2^*(k) \tag{15}$$

Step 4: Then obtain the correlation results $y_0(n)$ in the time-domain by applying IFFT to $Y_0(k)$. If the peak value of $y_0(n)$ exceeds the threshold value, the value of $Y_1(k)$ and $Y_2(k)$ will be computed according to the fusion algorithm shown in Equation (9). Then obtain the correlation results $y_1(n)$ and $y_2(n)$ in the time-domain by applying IFFT to $Y_1(k)$ and $Y_2(k)$.

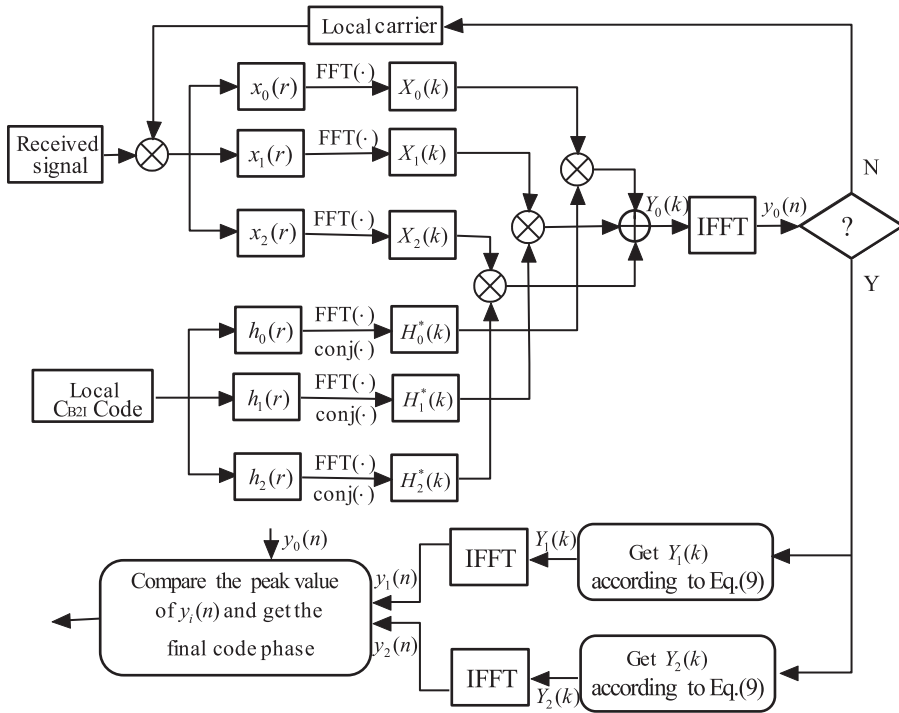


Figure 5. Acquisition process of 3-PCA algorithm.

The final code phase can be determined by comparing the peak values of $y_0(n)$, $y_1(n)$ and $y_2(n)$. Otherwise the frequency of the local carrier should be changed and Steps 1 to 4 need to be repeated until a correlation peak appears.

The proposed algorithm shortens the length of signals for FFT (Kurz et al., 2010) operation by dividing them into three parts, which makes it easy to achieve on an embedded platform with limited resources (Humphreys et al., 2006). Moreover, in the PCA algorithm (Leclère et al., 2013), the local C_{B2I} code shifts by only one bit in every correlation operation, while it shifts by three bits every time in the 3-PCA algorithm. In addition, the number of sampling points involved in every correlation operation remains the same in the two algorithms. Hence, the proposed algorithm will not decrease the acquisition accuracy.

4.2. *The principle of the K-PCA algorithm.* The above 3-PCA algorithm divides the mixed signal and local C_{B2I} code into three parts. Furthermore, the signal can be divided into K parts to perform the correlation operation. However, with the increase of the K value, the complexity of the algorithm will also be increased. The principle of the K -PCA algorithm is shown as follows.

First, the mixed signal is divided into K parts:

$$\begin{cases} x(rK) = x_0(r) \\ x(rK + 1) = x_1(r) \\ \vdots \\ x(rK + K - 1) = x_{K-1}(r) \end{cases} \quad r \in \left[0, \frac{N}{K} - 1 \right] \quad (16)$$

Then a FFT is applied to $x(n)$, and $X(k)$ can be written as Equation (17):

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n)W_N^{nk} = \sum_{r=0}^{N/K-1} x(rK)W_N^{rKk} + \sum_{r=0}^{N/K-1} x(rK+1)W_N^{(rK+1)k} \\
 &\quad + \dots + \sum_{r=0}^{N/K-1} x(rK+K-1)W_N^{(rK+K-1)k} \\
 &= \sum_{r=0}^{N/K-1} x_0(r)W_{N/K}^{rk} + W_N^k \sum_{r=0}^{N/K-1} x_1(r)W_{N/K}^{rk} \\
 &\quad + \dots + W_N^{(K-1)k} \sum_{r=0}^{N/K-1} x_{K-1}(r)W_{N/K}^{rk} = \sum_{i=0}^{K-1} W_N^{ik} X_i(k) \\
 X_i(k) &= FFT(x_i(n))
 \end{aligned}
 \tag{17}$$

Similarly, $H^*(k)$ and $Y(k)$ can be represented as:

$$H^*(k) = \sum_{i=0}^{K-1} W_N^{-ik} H_i^*(k), \quad Y(k) = \sum_{i=0}^{K-1} W_N^{ik} Y_i(k)
 \tag{19}$$

The correlation result in the frequency domain is shown in Equation (20):

$$Y(k) = X(k)H^*(k) = \left(\sum_{i=0}^{K-1} W_N^{ik} X_i(k) \right) \times \left(\sum_{i=0}^{K-1} W_N^{-ik} H_i^*(k) \right)
 \tag{20}$$

according to Equations (19) and (20), $Y_i(k)(i \in [0, K - 1])$ can be defined as follows

$$\begin{aligned}
 Y_0(k) &= H_0^*(k)X_0(k) + H_1^*(k)X_1(k) + \dots + H_{K-2}^*(k)X_{K-2}(k) + H_{K-1}^*(k)X_{K-1}(k) \\
 Y_1(k) &= H_0^*(k)X_1(k) + \dots + H_{K-3}^*(k)X_{K-2}(k) + H_{K-2}^*(k)X_{K-1}(k) + W_{N/K}^{-k} H_{K-1}^*(k)X_0(k) \\
 &\quad \vdots \\
 Y_{p-1}(k) &= H_0^*(k)X_{K-1}(k) + W_{N/K}^{-k} H_1^*(k)X_0(k) + W_{N/K}^{-k} H_2^*(k)X_1(k) + \dots + W_{N/K}^{-k} H_{K-1}^*(k)X_{K-2}(k)
 \end{aligned}
 \tag{21}$$

The correlation results in the time domain $y_i(n)$, therefore, can be obtained by applying an IFFT to $Y_i(k)$. The process of the correlation is the same as the process shown in Figure 5 where $K = 3$. In this algorithm, $h(n)$ shifts by K bits in every correlation operation. We can choose the value of K according to the situation. Meanwhile, the value of K should be smaller than the number of sampling points contained in half a C_{B2I} code chip. This is because when the local C_{B2I} code shifts more than half a chip every time, there will be no peaks in $y_i(n)$ even though the frequency of the received signal and local carrier is the same. In addition, the choice of K is also related to the number of sampling points N , which will be analysed in the next section.

4.3. *Method to reduce the computational complexity.* $y_0(n)$ is the correlation result between $h(n)$ and $x(n)$, where $h(n)$ shifts by K bits every time. In this acquisition process, if the peak value of $y_0(n)$ exceeds the threshold, $y_i(n)(i \in [1, K - 1])$ will be further

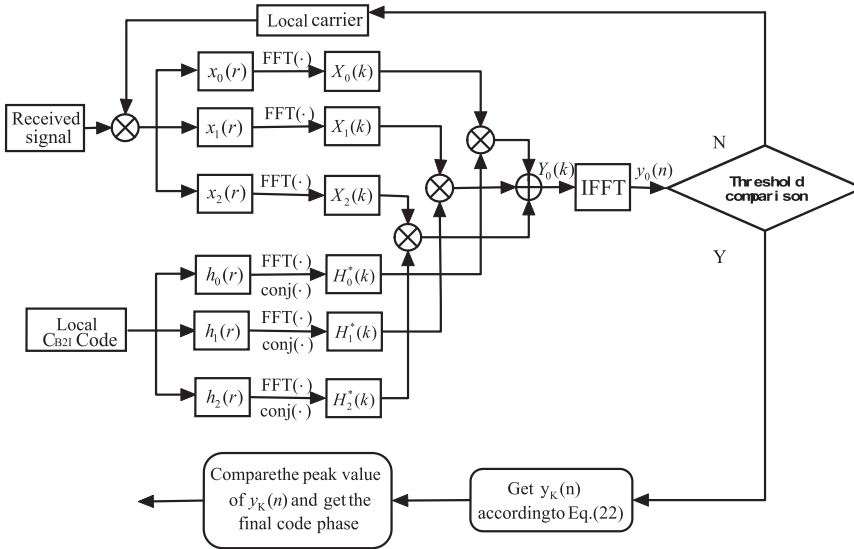


Figure 6. Process of the method to reduce the computational load.

computed to determine at which sampling point the code phase is located. This operation is not efficient enough as it involves a number of multiplications and IFFTs which are computation-intensive. Since the range code phase can be derived from the peak position of $y_0(n)$, the final code phase can be asserted by comparing the correlation values in the sampling points around the peak position of $y_0(n)$.

$h(n)$ shifts by K bits every time it is correlated with $x(n)$. If the coordinate of the peak of $y_0(n)$ is (m, R) , the range of the code phase should be $(K \times m - K \sim K \times m + K)$. Then the final code phase can be determined by comparing the correlation results at the $(K \times m - K \sim K \times m + K)$ th sampling point. The process is illustrated in Figure 6 (take $K = 3$ as an example) and Equation (22). This method only needs $2K \times N$ multiplications and $2K \times (N - 1)$ additions, and the procedure is simpler.

$$y_K(n) = \sum_{m=0}^{N-1} x(m)h(m - n) \quad n \in [K \times m - K, K \times m + K] \tag{22}$$

5. ANALYSIS OF COMPUTATIONAL COMPLEXITY. When implemented on an embedded platform with limited resources, the time complexity of an acquisition algorithm will affect the start-up speed of the software receiver. In a traditional hardware receiver, signal acquisition is achieved by an Application Specific Integrated Circuit (ASIC), whose computing speed is high. In a software receiver, signal acquisition is achieved by software running in the Central Processing Unit (CPU). It has a more flexible structure but a slower speed (Liu et al., 2011). Therefore, a software receiver will become more useful than a hardware receiver if we can reduce the time complexity of the acquisition algorithm.

The K -PCA algorithm decomposes the N -point FFT to K parts. The first issue to consider is the number of points of the FFT when the acquisition algorithm is implemented on

Table 1. Computational complexity comparison.

Step	PCA Algorithm		K-PCA Algorithm			
	Compute $y(n)$		Compute $y_0(n)$		Compute $y_i(n)$	
	MP	AP	MP	AP	MP	AP
Step 1	N	0	N	0	0	0
Step 2	$\frac{M}{2} \log_2^M$	$M \log_2^M$	$K \times \frac{F}{2} \log_2^F$	$K \times F \log_2^F$	0	0
Step 3	N	0	N	$\frac{N}{K} \times (K - 1)$	$\frac{3N}{2} \times (K - 1)$	$\frac{(K - 1)^2}{K} \times N$
Step 4	$\frac{M}{2} \log_2^M$	$M \log_2^M$	$\frac{F}{2} \log_2^F$	$F \log_2^F$	$(K - 1) \times \frac{F}{2} \log_2^F$	$(K - 1) \times F \log_2^F$
Sum	$y(n) \times 21$		$y_0(n) \times 21 + y_i(n)$			
Sum	MP	$21M \log_2^M + 42N$	$(11K + 10) \times F \log_2^F + \left(40 + \frac{3K + 1}{2}\right) \times N$			
	AP	$42M \log_2^M$	$(22K + 20)F \log_2^F + \left(19 + K - \frac{20}{K}\right)N$			

the hardware platform. If the algorithm uses the Cooley-Tukey radix-2 FFT structure, the FFT input vector needs to be fulfilled up to a power of two with zeros. If the split-radix FFT structure is chosen, this issue should not cause any concern (Tamazin et al., 2016; Molino et al., 2008). We will analyse the computational load in these two situations in the following parts.

5.1. *Time complexity analysis with radix-2 FFT structure.* The number of points the FFT needs to evaluate in the conventional PCA algorithm and K -PCA algorithm are shown in Equations (23) and (24) respectively:

$$\text{Length of FFT blocks in the PCA algorithm: } M = 2^{\text{ceil}(\log_2^N)} \tag{23}$$

$$\text{Length of FFT blocks in the } K\text{-PCA algorithm: } F = 2^{\text{ceil}(\log_2^{N/K})} \tag{24}$$

where N is the number of sampling points to be processed, and *ceil* represents the function that rounds the number up to the nearest integer.

In a common scenario, the Doppler frequency ranges between -10kHz to 10kHz . Assuming the scan step is 1kHz , the acquisition algorithm needs to search for the carrier frequency offset 21 times. If the frequency of the two signals (received signal and local carrier) is different, the K -PCA algorithm only needs to compute the result of $y_0(n)$. While the frequency of the two signals is the same, $y_i(n)(i \in [0, K - 1])$ will be calculated. The computational complexity of different algorithms is listed in Table 1.

The information related to Table 1 is as follows:

- Step 1: Strip the carrier off the received signal.
- Step 2: Apply FFT to $x(n)$ or $x_i(n)(i \in [0, K - 1])$. Due to the local C_{B2I} code, $h(n)$ is fixed during the algorithm operation process, the $H^*(k)$ and $H_i^*(k)(i \in [0, K - 1])$ are stored in memory without being calculated.
- Step 3: Multiply $X(k)$ with $H^*(k)$ to obtain $Y(k)$ or combine $X_i(k)(i \in [0, K - 1])$ with $H_i^*(k)(i \in [0, K - 1])$ together to obtain $Y_i(k)(i \in [0, K - 1])$.

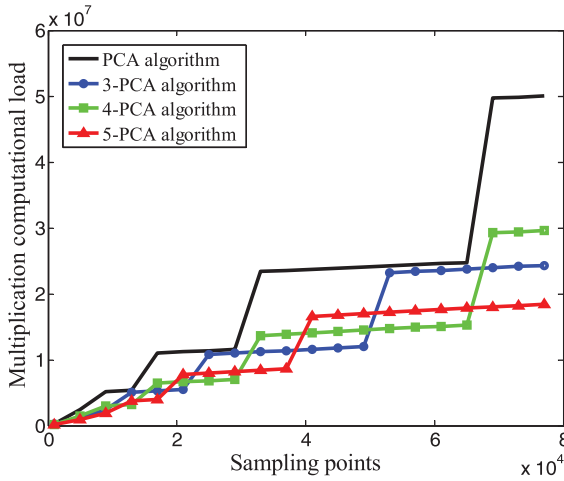


Figure 7. Comparison of computational load.

Step 4: Apply IFFT with the results obtained in Step 3.

MP: Multiplication operation.

AP: Addition operation.

With the increase of sampling points, the growing trend of computational load with different K values is shown in Figure 7.

If the K -PCA algorithm has the lowest computational load, we call the corresponding K the optimal K value. For example, if the computational load of 3-PCA is lowest, the optimal K is three. Provided that the sampling frequency and the number of sampling points are certain, we can choose the optimal K value to achieve the lowest computational complexity. In Figure 7, the optimal K value changes when the computational load has a step change. The parameter that can cause the step change of the computational load is F . Next, we will analyse its relationship with the optimal K value.

In Equation (24), $F = 2^{ceil(log_2^{N/K})}$, and by using t_K to represent the exponential part $ceil(log_2^{N/K})$, then $F = 2^{t_K}$. The change of F is related to the parameter t_K , so we will analyse the step change of t_K .

$$t_K = ceil(log_2^{N/K}) = ceil(log_2^N - log_2^K) \tag{25}$$

log_2^N and log_2^K can be presented as the forms in Equation (26), where D_N and D_K represent the decimal part of log_2^N and log_2^K respectively, and I_N and I_K represent the integer part.

$$log_2^N = \underbrace{I_N}_{\text{integer part}} \cdot \underbrace{D_N}_{\text{decimal part}} \quad log_2^K = \underbrace{I_K}_{\text{integral part}} \cdot \underbrace{D_K}_{\text{decimal part}} \tag{26}$$

If $D_N > D_K$, $t_K = ceil(log_2^N - log_2^K) = I_N - I_K + 1$; otherwise, $t_K = ceil(log_2^N - log_2^K) = I_N - I_K$.

When $D_N = D_K$, t_K will change suddenly and F will have a step change. Therefore, the optimal K value is related to the values of D_N and D_K . The length of the FFT block in different situations is shown in Table 2.

Table 2. Length of FFT block.

	$0 < D_N \leq D_5$	$D_5 < D_N \leq D_3$	$D_3 < D_N \leq 1$
$K = 1$	2^{I_N+1}	2^{I_N+1}	2^{I_N+1}
$K = 3$	$3 \times 2^{I_N-1}$	$3 \times 2^{I_N-1}$	3×2^{I_N}
$K = 4$	$4 \times 2^{I_N-1}$	$4 \times 2^{x_N-1}$	$4 \times 2^{x_N-1}$
$K = 5$	$5 \times 2^{x_N-2}$	$5 \times 2^{I_N-1}$	$5 \times 2^{I_N-1}$
Optimal K	$K = 5$	$K = 3$	$K = 4$

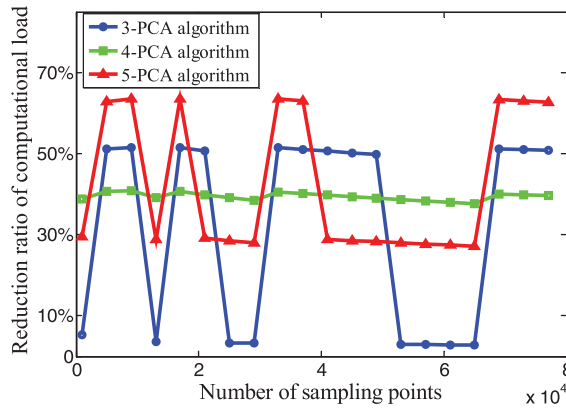


Figure 8. The reduction ratio of computational load.

By analysing Table 2 and Figure 7, we can see that the optimal K value can be determined by the relationship between D_N and D_K . If $D_{K_1} < D_N \leq D_{K_2}$, the optimal K value is K_2 . Since $D_{2K_2} = D_{K_2}$, $2K_2$ is also the optimal K value. However, as K increases, the algorithm will become more and more complex. Therefore, the K value should be selected according to the actual situation.

In summary, the optimal K value selection method is: 1. Calculate the value of l , which is the number of sampling points contained in half a C_{B2l} code chip. 2. Compare the values of D_N and D_K . If $D_{K_1} < D_N \leq D_{K_2}$, the optimal K value is $2^k K_2$ ($k \in [0, \log_2^l$, k is an integer). If there is more than one optimal value, the final optimal K value can be selected according to the structure complexity of the algorithm. As shown in Figure 8, the computational load can be reduced by as much as about 50% with an appropriate K value.

5.2. Time complexity analysis with split-radix FFT structure. There is no need to consider the zero-padding problem with the use of a split-radix FFT structure, and the computational load will also change. The Doppler frequency range is still set to -10 kHz to 10 kHz and the scan step is 1 kHz. The acquisition algorithm needs to search for the carrier frequency offset 21 times. The acquisition process is the same as the process shown in Figure 5. So, in the K -PCA algorithm, $y_0(n)$ should be computed 21 times, and $y_i(n)$ ($i \in [1, K - 1]$) only needs to be computed once. In the PCA algorithm, $y(n)$ should be computed 21 times. The computational complexities of the PCA algorithm and K -PCA algorithm are listed in Table 3. With the number of sampling points varying from 1×10^3 to 80×10^3 , the trend of the computational load of different algorithms is shown in Figure 9. Compared with the PCA algorithm, the K -PCA algorithm exhibits a lower computational burden. Figure 10 shows the reduction ratio of computational load with different numbers

Table 3. Computational complexity of two algorithms with split-radix FFT structure.

Step	PCA algorithm		The proposed algorithm decomposes the N-point FFT to P parts			
	Compute $y(n)$		Compute $y_0(n)$		Compute $y_i(n)$	
	MP	AP	MP	AP	MP	AP
Step 1	N	0	N	0	0	0
Step 2	$\frac{N}{2} \log_2^N$	$N \log_2^N$	$\frac{N}{2} \log_2^{\frac{N}{K}}$	$N \log_2^{\frac{N}{K}}$	0	0
Step 3	N	0	N	$\frac{N}{K} \times (K - 1)$	$\frac{3N}{2} \times (K - 1)$	$\frac{(K - 1)^2}{K} \times N$
Step 4	$\frac{N}{2} \log_2^N$	$N \log_2^N$	$\frac{N}{2K} \log_2^{\frac{N}{K}}$	$\frac{N}{K} \log_2^{\frac{N}{K}}$	$(K - 1) \times \frac{N}{2K} \log_2^{\frac{N}{K}}$	$(K - 1) \times \frac{N}{K} \log_2^{\frac{N}{K}}$
Sum	$y(n) \times 21$		$y_0(n) \times 21 + y_i(n)$			
Sum	MP	$21N \log_2^N + 42N$	$(81 + 3K)/2N + (11K + 10)/KN \log_2^{(N/K)}$			
	AP	$42N \log_2^N$	$(19 + K - 20/K)N + (22K + 20)/KN \log_2^{(N/K)}$			

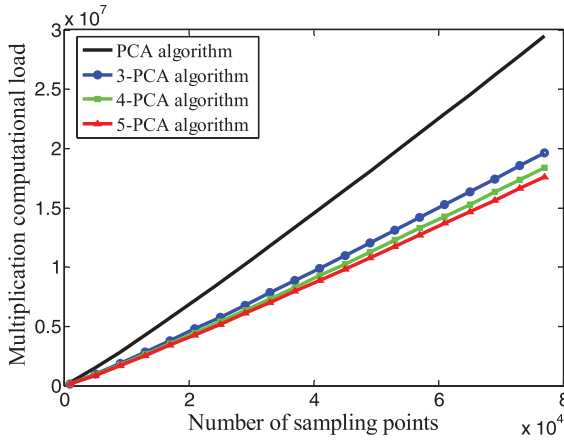


Figure 9. Multiplication computational load of algorithms with different K values.

Table 4. Parameters of BeiDou intermediate frequency signal.

Parameter name	BeiDou B2	Parameter name	BeiDou B2
Frequency	1207.140 MHz	C_{B2I} code length	2046 chips
IF signal frequency	4.5 MHz	Period	1 ms
Sampling frequency	18.073 MHz	Signal power	-160 dBW
C_{B2I} code rate	2.046 Mcps	Carrier to noise ratio	45 dB-Hz

of sampling points. Since the reduction ratio is proportional to the value of K , a larger K value can be chosen when acquisition accuracy is not affected.

6. SIMULATION RESULTS. To validate the performance of the K -PCA algorithm, we applied the algorithm to capture the B2 signal generated by MATLAB, which has been filtered, down-converted to an Intermediate Frequency (IF) and digitised to a digital signal. The related parameters are shown in Table 4.

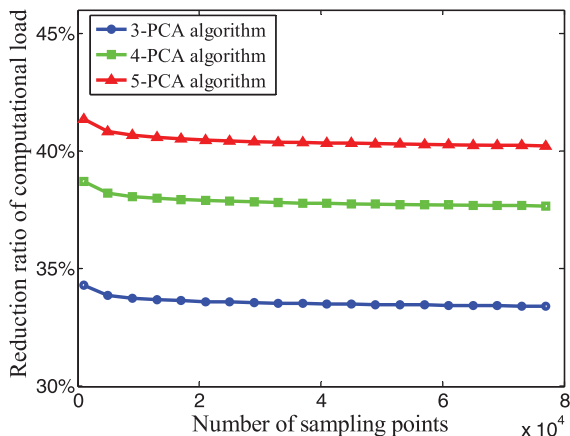


Figure 10. Reduction ratio of computational load of algorithms with different K values.

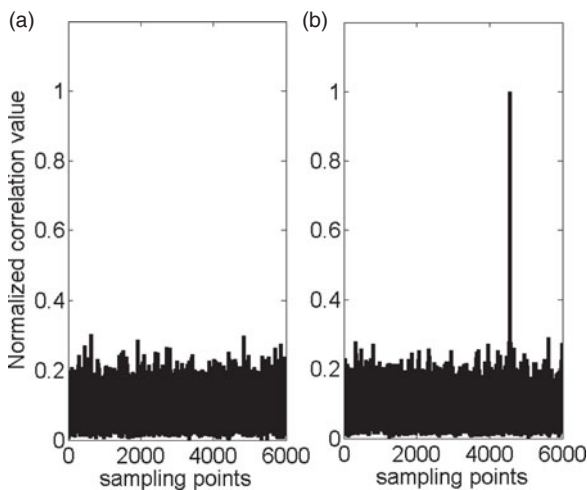


Figure 11. Results of $y_0(n)$ in different local carrier frequency situations.

From Table 4, we can see that the signal period is 1 ms and sampling frequency is 18.073 MHz, so the number of sampling points to be processed is 18,073. The number of sampling points contained in a half C_{B2I} code chip is four or five, so K should not exceed four. As $\log_2^{18073} = 14.1415$, $\log_2^3 = 1.585$ and $D_N < D_3$, the optimal k value should be three. Hence, we chose $K = 3$.

In this simulation process, a two-dimensional search for carrier frequency offset and code shift is carried out. The frequency offset range is set between -10 kHz and 10 kHz and is scanned with a 1 kHz step. The frequency offset is set to 2 kHz. The code phase is located at the 13,665th sampling point. If the frequencies of the received signal and local carrier are different, the result of $y_0(n)$ is similar to Figure 11(a). Accordingly, the frequency of the local carrier should be changed and the acquisition process repeated until a correlation peak appears. If the peak value in $y_0(n)$ is much larger than the others, as shown in Figure 11(b). This indicates that the search for carrier frequency offset and a

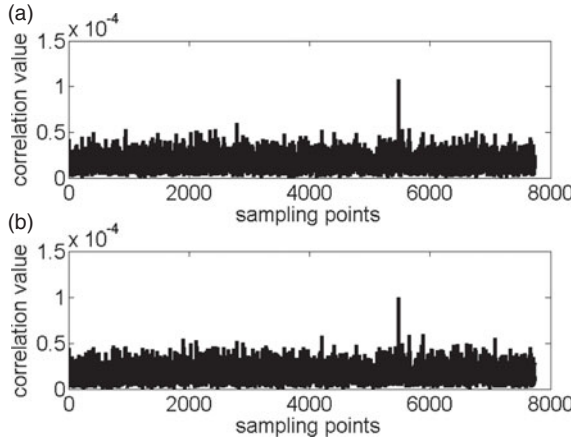


Figure 12. Results of $y_1(n)$ and $y_2(n)$.

rough estimation of code phase have been accomplished. At this stage, two methods can be used to find the final code phase. Method 1: As illustrated in Figure 5, the results of $y_1(n)$ and $y_2(n)$ are computed and their peak values are compared. If the largest peak value belongs to $y_i(n)$ and its coordinate is (m, R) , the final code phase is located at the $(3 \times m + i)$ th sampling point. Method 2: Use the methods shown in Figure 6 and Equation (22) to compute the correlation results in the sampling points near the peak position of $y_0(n)$. The code phase can be determined according to the position of the largest correlation result. The results of $y_1(n)$ and $y_2(n)$ in Method 1 are displayed in Figure 12, where the peak value coordinates of $y_0(n)$, $y_1(n)$ and $y_2(n)$ are $(4555, 1.253 \times 10^{-4})$, $(4554, 1.182 \times 10^{-4})$ and $(4554, 1.242 \times 10^{-4})$ respectively. So, the final code phase is located at the $(4555 \times 3 + 0 = 13663)$ th sampling point. Figure 13 shows the results of $y_K(n)$ obtained with Method 2. From the figure, we can see that the correlation result at the 13,665th sampling point is the largest. Accordingly, the code phase is located at the 13,665th sampling point. Obviously, the final code phase obtained by the two methods is the same as the pre-set value, so both methods can find the final code phase accurately. The Doppler frequency offset results are shown in Figure 14, and the frequency offset is 2 kHz which is the same as the pre-set value. In conclusion, the K -PCA algorithm can successfully estimate the Doppler shift and the code phase of the B2 signal.

From Section 5.1, we know that the value of K is related to the number of sampling points to be processed. When the signal sampling frequency is changed from 18.073 MHz to 15.491 MHz, the number of sampling points that need to be processed in one period is 15,491. The number of sampling points contained in half a C_{B2I} code chip is three to four, so K should not exceed three. As $\log_2^{15491} = 13.9191$, $\log_2^3 = 1.713$ and $D_3 < D_{15491} < D_2$, (Define $D_2 = 1$) the optimal K value should be two. Hence, we choose $K = 2$. In this simulation, the C/N_0 of the signal is 40 dB·HZ. The code phase and the Doppler frequency are set at the 1,446th chip and 2 kHz. The acquisition results of 2-PCA are as shown in Figures 15–18.

When the frequencies of the received signal and local carrier are different, the result of $y_0(n)$ is as Figure 15(a). If the received signal and local carrier frequencies are the same, the result of $y_0(n)$ is shown in Figure 15(b). The peak value in $y_0(n)$ is much larger than

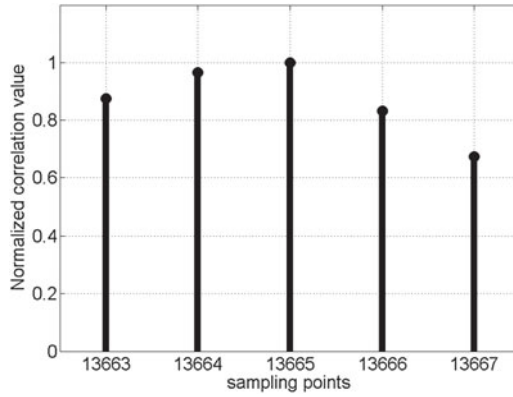


Figure 13. Results of $y_K(n)$.

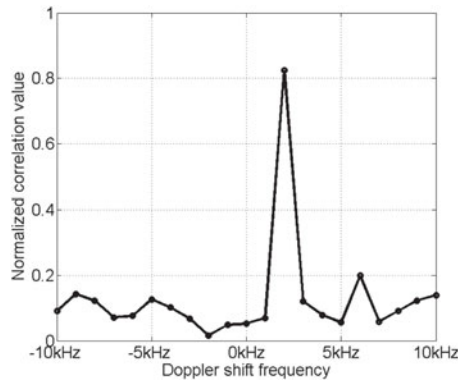


Figure 14. Doppler frequency offset map.

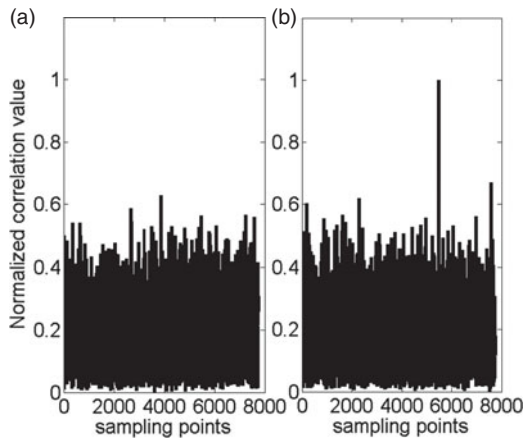


Figure 15. Results of $y_0(n)$ in different local carrier frequency situations.

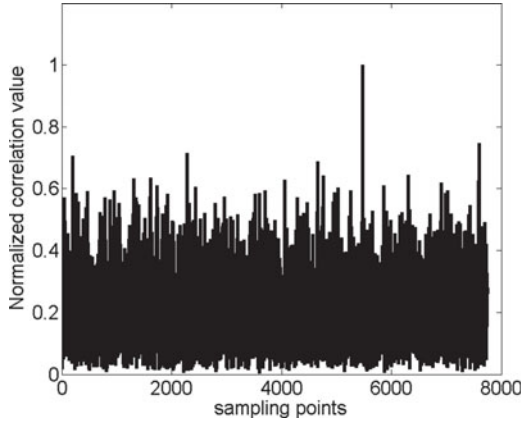


Figure 16. Results of $y_1(n)$.

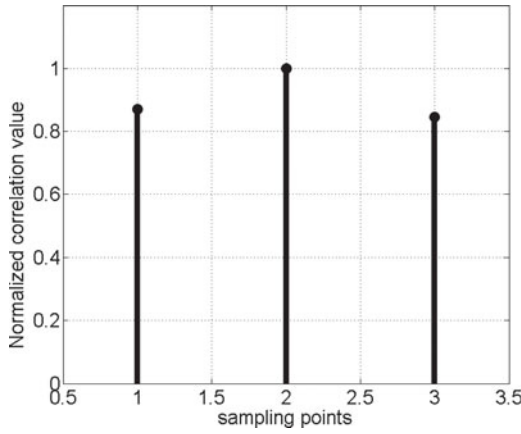


Figure 17. Results of $y_K(n)$.

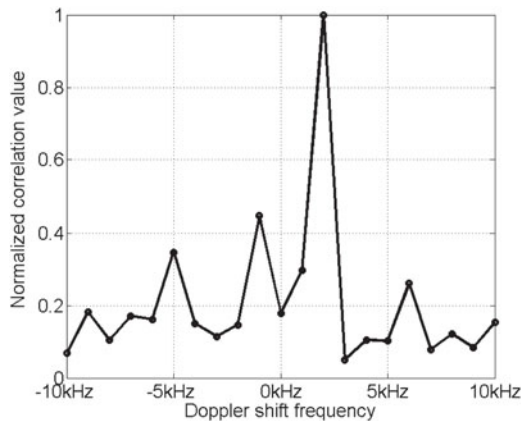


Figure 18. Doppler frequency offset map.

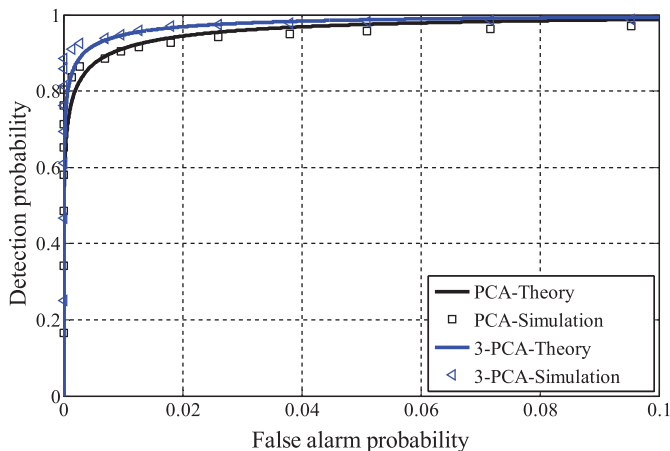


Figure 19. ROC comparisons between PCA and 3-PCA.

the others. Then the result of $y_1(n)$ and $y_K(n)$ should be calculated. The final code phase can be determined by comparing the peak values of $y_0(n)$ and $y_1(n)$, or by comparing the values of $y_K(n)$. Figures 16 and 17 show the results of $y_1(n)$ and $y_K(n)$. The peak value coordinates of $y_0(n)$, $y_1(n)$ are $(5471, 1.034 \times 10^{-4})$, $(5476, 9.7491 \times 10^{-5})$ respectively. So, the final code phase is located at the $(5471 \times 2)/15491 \times 2046 \approx 1446$ th chip. From Figure 3, we can see that the peak value of $y_K(n)$ is located at the 10,952nd sampling point. So, the final code phase is located at the $10953/15491 \times 2046 \approx 1446$ th chip. The two methods have the same acquisition results. The Doppler frequency offset results are shown in Figure 18, and the frequency offset is 2 kHz, which is the same as the pre-set value.

For the SSA algorithm, when the code phase search step changes, its acquisition results still obey the Rician distribution. Compared with the PCA algorithm, the acquisition results of K -PCA are still the cross-correlation results between $x(n)$ and $h(n)$, and the difference is that $h(n)$ shifts by three bits in every cross-correlation operation. The code phase search step changes, but the acquisition results of K -PCA should still obey the Rician distribution. The threshold values of PCA and K -PCA can be calculated according to Equation (27). In this equation, P_{fa} is the probabilities of false alarm, σ is the root value of noise power. σ^2 can be calculated according to Equation (28), where $V(n)$ is the acquisition value in one search bin and M is the number of search bins.

$$V_T = \sigma \sqrt{-2 \ln(P_{fa})} \tag{27}$$

$$\sigma^2 = \frac{1}{2M} \sum_{n=1}^M V^2(n) \tag{28}$$

To assess the performance of the K -PCA, Monte Carlo simulation is performed to the Receiver Operating Characteristic (ROC) curve (Yang et al., 2014; 2016). ROC represents the detection probability under different probabilities of false alarm. Figure 19 shows ROC curves of PCA and 3-PCA. From Figure 19 we can see that the detection probabilities under the same P_{fa} of 3-PCA is slightly lower than that of PCA. This is because the code

search steps of the two algorithms are different, and the peak value of the 3-PCA algorithm is slightly lower than that of the PCA algorithm.

7. CONCLUSION. A novel acquisition algorithm is proposed in this paper. Unlike the PCA algorithm, which obtains correlation results at every sampling point, our algorithm does not evaluate correlation results at every sampling point but at intervals of K points. The K value can be arbitrarily selected according to the sampling frequency and the number of sampling points to be processed. If the K value is optimal, the computational load can be reduced by about 50%; otherwise, its detection probability is slightly lower than that of the PCA algorithm, under the same probability of false alarm. Therefore, this algorithm can be used to achieve fast acquisition when the power of the signal is weak and the sampling frequency is high or the number of sampling points to be processed is large.

FINANCIAL SUPPORT

This work was supported by the National Natural Science Foundation of China (51505221).

REFERENCES

- Akopian, D. (2005). Fast FFT based GPS satellite acquisition methods. *IET Radar Sonar Navigation* 2005, **152**(4), 277–286.
- Cheng, P.Q. (2012). *Digital signal processing (Fourth Edition)*. Tsinghua University Press, 139–171.
- China Satellite Navigation Office. (2013). *BeiDou navigation satellite signal in space interface control document open service signal. Version 2.0*. Beijing: China Standardization, 5–25.
- Chun, Y., Mikel, M. and Thao, N. (2006). Generalized Frequency-Domain Correlator for Software GPS Receiver: Preliminary Test Results and Analysis. *19th International Technical Meeting of the Satellite Division of the Institute-of-Navigation*. 26–29 September, Fort Worth, 2346–2360.
- Chun, Y. (2005). Joint acquisition of CM and CL codes for GPS L2 civil (L2C) signals. *Proceedings of the ION 61st Annual Meeting*. 27–29 June 2, Cambridge, 553–562.
- Fantino, M., Pini, M., Mulassano, P., Girau, G., Nicola, M. and Nordio, A. (2008). Signal Compression for an Efficient and Simplified GNSS Parallel Acquisition. *ION GNSS 21st International Technical Meeting of The Satellite Division*, 16–19 September, Savannah, 159–166.
- Humphreys, T.E., Psiaki, M.L. and Kintner, P.M. (2006). GNSS receiver implementation on a DSP: status, challenges, and prospects. *GNSS 19th International Technical Meeting of the Satellite Division*. 26–29 September, Fort Worth, 2370–2382.
- Jan, S.S. and Lin, Y.C. (2009). A new multi-C/A code acquisition method for GPS. *GPS Solutions*, **13**, 293–303.
- Jin, T., Yang, J., Huang, Z. and Qin, H. (2015). Multi-correlation strategies fusion acquisition method for high data rate global navigation satellite system signals. *IET Signal Processing*, **9**(8), 623–630.
- Jin, T., Lu, F., Liu, Y., Qin, H. and Luo, X. (2013). Double differentially coherent pseudorandom noise code acquisition method for code-division multiple access system. *IET Signal Processing*, **7**(7), 587–597.
- Jin, T., Ye, W., Lin, S. and Hua, Z. (2008). Software Defined Radio GNSS Receiver Design over Single DSP Platform. *4th International Conference on Wireless Communications, Networking and Mobile Computing*. 12–17 October, Dalian, 1423–1426.
- Kurz, L., Kappen, G., Coenen, T. and Noll, T. G. (2010). Comparison of Massive-Parallel and FFT-Based Acquisition Architectures for GNSS Receivers. *Proceedings of International Technical Meeting of the Satellite Division of the Institute of Navigation*. September, Portland, USA, 2874–2883.
- Leclère, J., Botteron, C. and Farine, P.A. (2013). Modified parallel code-phase search for acquisition in presence of sign transition. *International Conference on Localization and GNSS*. IEEE, 1–6.
- Leclère, J., Botteron, C. and Farine, P.A. (2012). Improving the Performance of the FFT-based Parallel Code-phase Search Acquisition of GNSS signals by Decomposition of the Circular Correlation. *25th International Meeting of the Satellite Division of the Institute of Navigation ION*. 12–17 September, Nashville TN, 1406–1416.

- Lin, D.M., Tsui, J.B.Y. and Howell, D. (1999). Direct P(Y)-Code Acquisition Algorithm for Software GPS Receivers. *12th International Meeting of the Satellite Division of the Institute of Navigation ION*. 14–17 September, Nashville, TN.
- Liu, Y., Jin, T. and Qin, H. (2011). A Real Time High Sensitive Software GPS Receiver Architecture and Verification. *International Technical Meeting of the Institute of Navigation*. 24–26 January, San Diego, 1246–1256.
- Molino, A., Girau, G., Nicola, M., Fantino, M. and Pini, M. (2008). Evaluation of a FFT-based Acquisition in Real Time Hardware and Software GNSS Receivers. *IEEE International Symposium on Spread Spectrum Techniques & Applications*. 25–28 August, Bologna, 37–41.
- Patel, V. and Shukla, P. (2011). Faster Methods for GPS Signal Acquisition in Frequency Domain. *2011 International Conference on Emerging Trends in Networks and Computer Communication*. 22–24 April, Udaipur, 84–88.
- Polydoros, A. and Weber, C.L. (1984). A unified approach to serial search spread-spectrum code acquisition-part I and part II. *IEEE Transactions on Communications*, **60**(1), 542–549.
- Simone, L., Fittipaldi, G. and Sanchez, I.A. (2011). Fast acquisition techniques for very long PN codes for on-board secure TTC transponders. *Proceedings of Military Communications Conference*, 7–10 November, Baltimore, 1748–1753.
- Starzyk, J.A. and Zhu, Z. (2001). Averaging Correlation for C/A code Acquisition and Tracking in Frequency Domain. *Proceedings of the 44th IEEE 2001 Midwest Symposium on Circuits And systems*. 14–17 August, Fairbom, 905–908.
- Tamazin, M., Noureldin, A., Korenberg, M.J. and Massoud, A. (2016). Robust fine acquisition algorithm for GPS receiver with limited resources. *GPS Solutions*, **20**, 77–88.
- Tsui, J.B.Y. (2005). *Fundamentals of global positioning system receivers—a software approach*. Wiley-Interscience, 155–280.
- Van Nee, D. and Coenen, A. (1991). New fast GPS code-Acquisition technique using FFT. *Electronics Letters*, **27**(2), 158–160.
- Wang, Z., Hu, J., Xiong, X. and Song, J. (2013). Non-data-aided timing acquisition for asynchronous IDMA systems. *Wireless Personal Communications*, **69**(2), 957–978.
- Xie, F., Liu, J., Li, R. and Feng, S. (2016). A simultaneous multiple BeiDou signal acquisition algorithm for a software-based GNSS receiver. *Optik*, **127**(4), 1607–1614.
- Yang, J., Jin, T., Huang, Z. and Qin, H. (2014). Multi-signal components combining acquisition method based on padding zero for TD-AltBOC. *Journal of Harbin Engineering University*, **35**(11), 1427–1433.
- Yang, J., Jin, T., Huang, Z. and Qin, H. (2016). Data and pilot optimised combining method for new composite global navigation satellite system signal acquisition. *IET Radar Sonar Navigation*, **10**(5), 953–965.
- Zhang, P., Xu, C., Cai, X. and Li, H. (2016). Performance analysis of BDS for regional services with consideration on weighted factors. *Proceedings of the Institution of Mechanical Engineers, Part G-Journal Aerospace Engineering*, **230**(1), 146–156.