# A Fast Acquisition Algorithm Based on High Sampling Rate FFT for LEO Satellite Signals

Deyang Kong(✉), Zhisong Hao, and Yongfei Hou

The 54th Research Institute of China Electronics Technology Group Corporation, 589 Zhongshan West Road, Shijiazhuang, China

**Abstract.** In the communication of LEO satellite system, there is a high relative velocity and acceleration between the receiver and the transmitter due to the high-speed motion of the platform. This leads to doppler frequency offset in the communication link. In large frequency offset and high dynamic communication environment, signal frame capture is the key technology to realize signal demodulation and link establishment. The acquisition bandwidth of low symbol rate signal is very small, which leads to the long acquisition time of frequency sweeping method. In this paper, the fast acquisition problem of low-speed FDMA signal with large frequency offset and high dynamic is solved by using high sampling rate partially matched filter FFT algorithm. The algorithm can realize the frequency estimation and fast acquisition of the signal when the frequency offset is 4 times the symbol rate. Compared with the traditional method, the algorithm greatly improves the acquisition bandwidth and speed.

**Keywords:** LEO satellite communication · Fast signal acquisition · Frequency offset estimation

## 1 Introduction

In recent years, with the application of LEO satellite constellation and various micro satellites, LEO satellite communication has been widely used [1]. However, compared with geostationary satellites, the high-speed motion of LEO satellite brings large Doppler frequency offset [2]. For example, when the symbol rate is more than 10 kbps, the Doppler frequency offset is as high as tens of kHz, which is far beyond the symbol rate, so it is difficult to capture the signal. This leads to a very bad communication environment for low-speed satellite signals. The low-speed FDMA signal of the geostationary orbit (GEO) is usually captured in a frequency sweep mode, and then the frequency offset is estimated through the pilot frequency [3]. However, the frequency offset estimation range of this method is half of the pilot rate, and the sweep step is related to the symbol rate and the length of the capture frame header, which results in a longer sweep time when the symbol rate is low [4]. This paper proposes a fast capture


Q. Guo et al. (Eds.): WiSATS 2021, LNICST 410, pp. 14–22, 2022.
https://doi.org/10.1007/978-3-030-93398-2_2

algorithm, which realizes the fast capture of the signal by means of frequency estimation and search on the frame header at different sampling rates. MATLAB simulation verifies that the algorithm can work normally when the signal noise ratio (SNR) is 3dB and the frequency offset range is 4 times the symbol rate.

## 2 Principle

This algorithm is mainly used for the fast acquisition of low symbol rate FDMA signals under large frequency offset. This algorithm estimates the frequency offset exceeding the symbol rate by performing frequency when the signal sampling rate is 8 times, then estimates the remaining frequency offset when the signal sampling rate is 1 times, and outputs the frame header position to achieve frame capture. The main process of this algorithm is shown in Fig. 1.

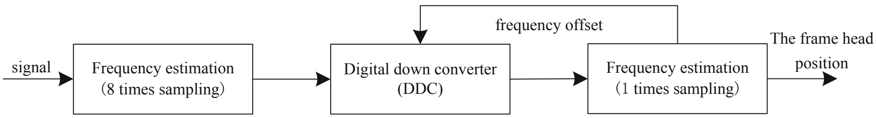


**Fig. 1.** The algorithm flow chart

The basic principles of frequency estimation are as follows.

Assuming that S(t) is the signal to enter Analog Devices (AD), then the S(t) can be expressed as

$$S(\mathrm{t}) = \mathrm{d(t)e}^{j(\omega_0 + f_0)t + \theta_0} + n(t) \tag{1}$$

In Eq. (1), $\omega_0$ is the carrier frequency; $f_0$ is the frequency offset; $\theta_0$ is the Signal phase; $n(t)$ is Gaussian white noise; $d_t$ represents the transmitted data; Within one frame

$$\mathrm{d(t)} = \begin{cases} Pn_i(t) + j \cdot Pn_q(t) & 0 \le \mathrm{t} \le 64\mathrm{T}_s \\ d_i(t) + j \cdot d_q(t) & 64\mathrm{T}_s < \mathrm{t} \le m\mathrm{T}_s \end{cases} \tag{2}$$

Ts is the symbol period, m is the number of symbols per frame. The format of the transmitted data frame is that the first 64 data is a fixed pseudo-random sequence, and the following is the data to transmit.

The signal S(n) sampled by AD can be expressed as

$$S(\mathrm{n}) = \sum_{n=0}^{n=N} \delta(nT_c) \cdot \left[\mathrm{d(t)} \cdot \mathrm{e}^{j(\omega_0 + f_0)t + \theta_0} + n(t)\right] = \begin{cases} \sum_{n=0}^{n=64*T_s/T_c} \delta(nT_c) \cdot \left[(Pn_i(t) + j \cdot Pn_q(t)) * \mathrm{e}^{j(\omega_0 + f_0)t + \theta_0} + n(t)\right] \\ \sum_{n=64*T_s/T_c + 1}^{n=mT_s/T_c} \delta(nT_c) \cdot \left[(d_i(t) + j \cdot d_q(t)) * \mathrm{e}^{j(\omega_0 + f_0)t + \theta_0} + n(t)\right] \end{cases} \tag{3}$$

In Eq. (2), $T_c$ is the sampling period and $T_s/T_c$ is an integer; N is the number of sampling points. Assuming that the signal entering ad is a baseband signal, then the sampled signal is S(n).

$$S(\mathrm{n}) = \begin{cases} \sum_{n=0}^{n=64*T_s/T_c} \delta(nT_c) \cdot \left[(Pn_i(t) + j \cdot Pn_q(t)) * \mathrm{e}^{jf_0 t + \theta_0} + n(t)\right] \\ \sum_{n=64*T_s/T_c + 1}^{n=mT_s/T_c} \delta(nT_c) \cdot \left[(d_i(t) + j \cdot d_q(t)) * \mathrm{e}^{jf_0 t + \theta_0} + n(t)\right] \end{cases} \tag{4}$$

$PN(\mathrm{n})$ is the local sequence generated by the PN sequence generator.

$$PN(\mathrm{n}) = \sum_{n=0}^{n=64T_s/T_c} \delta(nT_c) \cdot \left[Pn_i(t) + j \cdot Pn_q(t)\right] \tag{5}$$

$S_1(p)$ is the result of the conjugate multiplication of $S(\mathrm{n})$ and $PN(\mathrm{n})$.

$$S_1(p) = \sum_{p=0}^{p=(m-64)\cdot T_s/T_c} \left( \sum_{n=p}^{n=(64T_s/T_c)+p} S(n) \cdot PN(n) \right) \tag{6}$$

When p is 0, that is, when the local code and the random sequence in the selected signal are exactly aligned, the S1 can be expressed as

$$
\begin{aligned}
S_1 &= \sum_{n=0}^{64*T_s/T_c} \delta(nT_c) \cdot \left\{ \left[Pn_i(t) + j \cdot Pn_q(t)\right] \cdot \mathrm{e}^{jf_0 t + \theta_0} + n(t) \right\} \cdot PN(n) \\
&= \sum_{n=0}^{64*T_s/T_c} \delta(nT_c) \cdot \left\{ \left[Pn_i(t) + j \cdot Pn_q(t)\right] \cdot \mathrm{e}^{jf_0 t + \theta_0} + n(t) \right\} \cdot \left[Pn_i(t) - j \cdot Pn_q(t)\right] \\
&= \sum_{n=0}^{64*T_s/T_c} 2\delta(nT_c) \left\{ \left[1 - j \cdot Pn_i(t) Pn_q(t)\right] \right\} \cdot \mathrm{e}^{jf_0 t + \theta_0} + n(t) \cdot \left[Pn_i(t) - j \cdot Pn_q(t)\right] \\
&= \sum_{n=0}^{64*T_s/T_c} 2\left\{ \left[1 - j \cdot Pn_i(nT_c) Pn_q(nT_c)\right] \right\} \cdot \mathrm{e}^{jnf_0 T_c + \theta_0} + n(nT_c)
\end{aligned}
\tag{7}
$$

The $64T_s/T_c$ points fast Fourier transform is $S_1$ [5]. Then $S_1$ can be expressed as

$$
\begin{aligned}
S_1(k) &= \sum_{n=0}^{N-1} S_1 \mathrm{e}^{-i2\pi kn/N} \\
&= \sum_{n=0}^{N-1} \sum_{m=0}^{64*T_s/T_c} 2\left\{ \left[1 - j \cdot Pn_i(mT_c) Pn_q(mT_c)\right] \right\} \cdot \mathrm{e}^{jmf_0 T_c + \theta_0} \cdot \mathrm{e}^{-i2\pi kn/N}
\end{aligned}
\tag{8}
$$

When $\sum_{m=0}^{64*T_s/T_c} Pn_i(mT_c) Pn_q(mT_c) \approx 0$, we can get

$$
S_1(k) = \sum_{m=0}^{64*T_s/T_c} 4\pi \cdot e^{\theta} \cdot \delta(\mathrm{k} - mf_0 T_c) \tag{9}
$$

When $k = mf_0 T_c$, we can get $\max(S_1(k)) = 4\pi e^{\theta}$. The frequency offset value in the signal can be calculated according to the position of the maximum value. Because of the high-speed movement of the platform where the transmitter is located, there is a Doppler shift in the received signal. Assuming that the range of frequency offset is $[-f_{\max}, f_{\max}]$; Doppler rate of change is $a$; Doppler model is cosine model; we can get

$$
f(t) = f_{\max} \cdot \cos(\omega t) \tag{10}
$$

Derivative of $f(t)$ is $f'(t) = f_{\max} \cdot \omega \sin(\omega t)$. When $t = 0$, the change rate reaches the maximum.

$$
a = f_{\max} \cdot \omega \tag{11}
$$

So we can get

$$f(t) = f_{\max} \cdot \cos(\frac{at}{f_{\max}}) \tag{12}$$

The frequency offset of $64T_s/T_c$ signal sampling points is

$$\begin{aligned} f(n) &= \sum_{n=0}^{64*T_s/T_c} f(t) \cdot \delta(nT_c) \\ &= \sum_{n=0}^{64*T_s/T_c} f_{\max} \cos(\frac{at}{f_{\max}}) \cdot \delta(nT_c) \end{aligned} \tag{13}$$

When $a = 0$, the frequency offset of the intercepted signal is a fixed value. There is only one larger value in $S_1(k)$, and the remaining values are very small. When $a \neq 0$, the frequency component of the intercepted signal is not single, and there are many larger values of $S_1(k)$. Define $\lambda$

$$\lambda(m) = \begin{cases} 10 * \log_{10}(\frac{S_m(p) + S_m(p+1) + S_m(64 * T_s/T_c)}{\left[\sum_{n=1}^{64*T_s/T_c} S_m(n)\right] - S_m(1) - S_m(2) - S_m(64 * T_s/T_c)}) & \text{p = 1} \\ 10 * \log_{10}(\frac{S_m(p) + S_m(p-1) + S_m(p+1)}{\left[\sum_{n=0}^{64*T_s/T_c} S_m(n)\right] - S_m(p) - S_m(p-1) - S_m(p+1)}) & 1 < \text{p} < 64T_s/T_c \\ 10 * \log_{10}(\frac{S_m(1) + S_m(p-1) + S_m(p)}{\left[\sum_{n=1}^{64*T_s/T_c} S_m(n)\right] - S_m(1) - S_m(p-1) - S_m(p)}) & \text{p} = 64T_s/T_c \end{cases} \tag{14}$$

In Eq. (13), $m$ is the initial position of the intercepted signal; $S_m(n)$ is the result of $S_1(k)$ when the initial position of the intercepted signal is $m$; $p$ is the position of the maximum value in $S_m(n)$. Define the detection is successful when $\lambda \geq 15$. When $T_c = T_s$, the frequency resolution of the Fourier transform is $f_s$. Representative frequency range is $[-f_s/2, f_s/2]$. We can get $f_0 = (p - 32)f_s/128$. The error of frequency offset estimation is $f_s/128$. When $T_c = 8T_s$, the frequency resolution of the Fourier transform is $8f_s$. Representative frequency range is $[-4f_s, 4f_s]$. We can get $f_0 = (n - 256)f_s/128$.

## 3 Simulation

In this simulation, the rate of symbols is 3.29Ksys and the frequency offset is 5 kHz. The framer in the signal is 64 bit PN sequence and the length of FFT used in the simulation is 512.

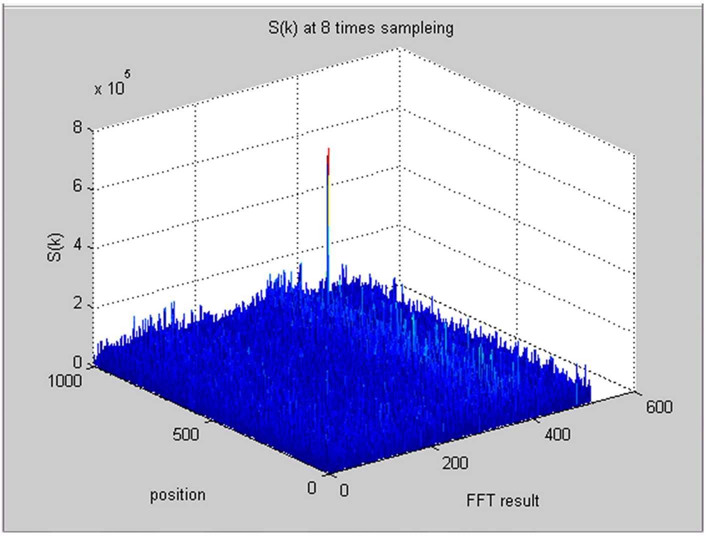


**Fig. 2.** S(k) simulation result at 8 times sampling

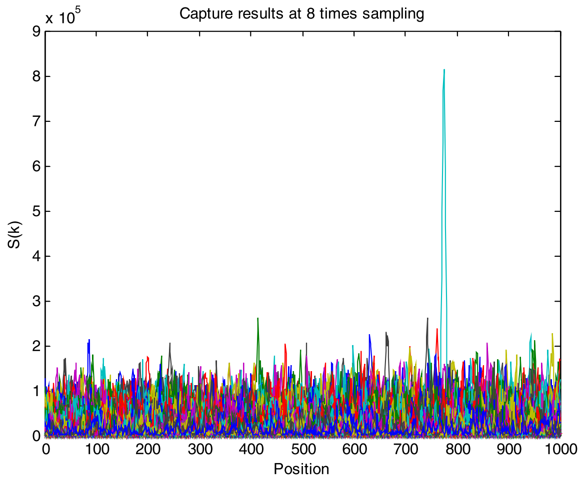


**Fig. 3.** S(k) cross-sectional view on the x-axis

Figure 2 is the result of S(k) at 8 times sampling. Through the Fig. 2 we can see that S(k) is very big at some points. Thus points' position represent the framer's position and the frequency offset.

Figure 3 and Fig. 4 are cross-sectional views of the capture results on the x-axis and y-axis at 8 times sampling. As shown in Fig. 3, the peak is the capture position, that is, the synchronization position of the signal. Figure 4 shows the position of the peak in the result of an FFT, which is p in Eq. 14. The position of this point represents

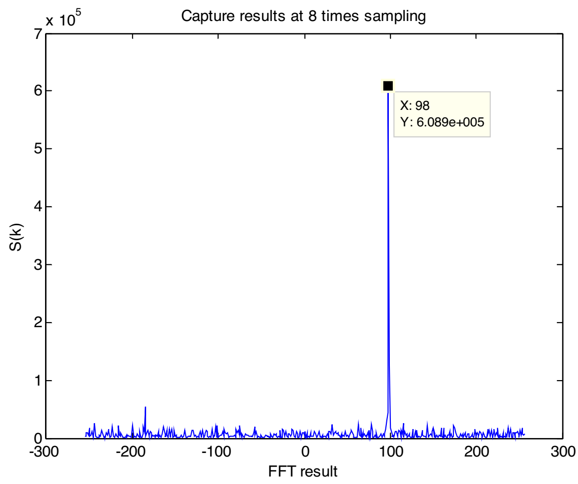


**Fig. 4.** S(k) cross-sectional view on the y-axis

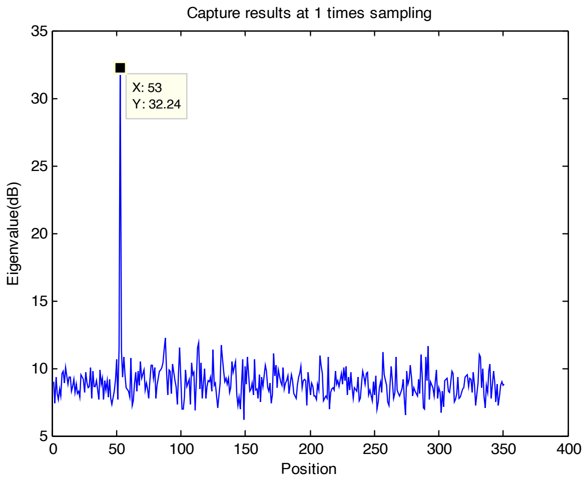


**Fig. 5.** Capture results at 1 time sampling

the frequency offset. From the figure, it can be seen that the peak position is 98. So we can get the estimated value of the frequency offset is 5.037 kHz.

Figure 5 shows the capture result of 1x sampling, and the captured position is the frame synchronization position. As shown in the figure, the calculation result at the 53rd sample point is significantly larger than other positions, which means that this point is a synchronous position.

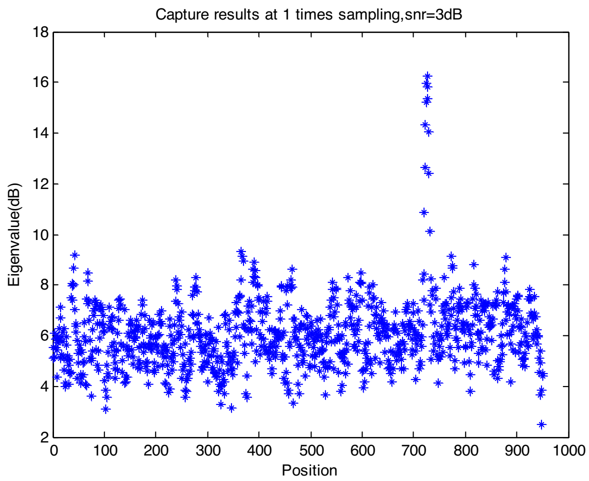


**Fig. 6.** Capture results when the SNR is 3 dB

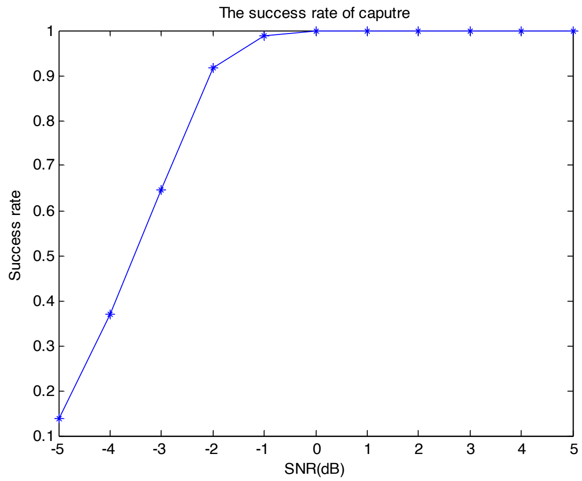


**Fig. 7.** Acquisition success rate at different SNR

Figure 6 shows the capture result when the signal-to-noise ratio is 3dB. The frequency offset can be estimated from the 8x sampling capture result, and the precise synchronization position can be captured when 1x sampling. Figure 7 shows the capture success rate at different SNR. At each SNR, 1000 acquisition simulations are carried out, and finally calculate the acquisition success rate.

## References

1. Tawk, Y., Jovanovic, A., Leclere, J., et al.: A new FFT-based algorithm for secondary code acquisition for galileo signals. In: Vehicular Technology Conference. IEEE (2011)
2. Sun, G., Huang, Q., Zhu, L.: A fast acquisition algorithm based on FFT for DSSS signal and FPGA realization. IEEE Computer Society (2009)
3. You, Y.H., Song, H.K.: Efficient sequential detection of carrier frequency offset and primary synchronization signal for 5G NR systems. IEEE Trans. Veh. Technol. **PP**(99), 1 (2020)
4. Venkataramanan, R., Prabhu, K.: Estimation of frequency offset using warped discrete-Fourier transform. Signal Process. **86**(2), 250–256 (2006)
5. Wang, Y., Zhang, X., Zheng, K.: An improved PMF-FFT algorithm based on all phase preprocessing. J. Telemet. Track. Command **39**, 16–20 (2018)