

RESEARCH

Open Access



# Research on multi-constellation GNSS compatible acquisition strategy based on GPU high-performance operation

Chengjun Guo<sup>1\*</sup> , Bingyan Xu<sup>2</sup> and Zhong Tian<sup>2</sup>

## Abstract

With the continuous development of satellite navigation, how to make full use of the compatibility and interoperability among the four constellations deserves our deep thinking. Signal acquisition is a critical technology that affects the performance of GNSS receivers. However, its implementation has high requirements on both resource consumption and processing time. In recent years, graphics processing unit (GPU) technology with a large number of parallel processing units is gradually applied to the navigation field. In this paper, a multi-constellation-compatible capture strategy based on GPU is proposed. Compared with the traditional GNSS signal processing, the design has better flexibility and portability, and the acquisition parameters of the system can be configured flexibly through the interface. The experimental results show that the proposed scheme makes full use of the powerful parallel processing capability of GPU, which greatly reduces the acquisition time and improves the efficiency of signal processing, with the increase of data processing. In addition, as the amount of signal processing data increases, the advantages of the CPU high-performance computing platform will be more obvious.

**Keywords:** Global navigation satellite system (GNSS), Graphics processing unit (GPU), Compatible, Signal acquisition

## 1 Introduction

Recently, GNSS applications will face a revolution due to the construction and modernization of global navigation satellite systems (GNSS), and research on multi-system and multi-frequency compatible receivers has become the trend. Integrated navigation technology can greatly improve the reliability and the precise of navigation system [1]. For the GNSS multi-constellation navigation system, because the four satellite navigation systems (GPS, GLONASS, Galileo signal, and Beidou) have different signal characteristics, including modulation mode, carrier frequency, and the generation of pseudo random noise code (PRN) code, making the consideration between the four satellite systems compatibility and interoperability is very important in the design of

the GNSS receiver [2]. Interoperability refers to the use of the same receiver system without any hardware modifications and only needs to set up software to receive signals from the four systems at the same time.

Signal processing is the core part of GNSS receiver, which is mainly composed of signal capture and signal tracking. In the initial stage of the development of the receiver, it is mainly implemented by hardware. Then, the technology of digital signal processing (DSP), field programmable gate array (FPGA), and advanced RISC machines (ARM) have been gradually applied to the design of the signal sampling or processing part of the GNSS software receiver [3–8]. Compared to the traditional hardware receiver using application-specific integrated circuit (ASIC), FPGA or DSP offer flexibility similar to software and speed similar to hardware, but there are

\* Correspondence: [15882000306@163.com](mailto:15882000306@163.com)

<sup>1</sup>National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China  
Full list of author information is available at the end of the article

difficulties in implementing high-complexity calculations, and they are still expensive niche products for dedicated applications [9, 10]. In addition, the flexibility of receivers based on FPGA, DSP, or FPGA+DSP is still very limited. At the same time, the central processing unit (CPU) has gradually been used in the design of the signal processing part of the GNSS software receiver [11]. But limited by the framework of CPU's own structure, the efficiency of FFT processing is not ideal, which makes it difficult to realize real-time processing.

Until recently, the graphics processing unit (GPU) technology with high-performance parallel computing has gradually developed [12–14]. GPU is very different from a CPU dedicated to general sequential tasks, which has the characteristics of high parallel number and large data throughput. GPU includes hundreds to thousands of special purpose processors for graphic processing and aims to carry out programming so that the same function can be performed on multiple data [15]. With the continuous development of GPU technology, its application range extends from the graphic field to more high-performance computing areas that require very strong computing power [16, 17]. The GPU has been expected to be another way for implementation which allows to realize the GNSS radio [18]. In [19], a novel GPU-based correlator architecture for GNSS software receivers is proposed. In [20], scholars have studied the GPU-based real-time GPS software receiver. It can be seen that the existing research results prove that GPU technology can be applied to the signal processing part of a software receiver and bring the improvement of signal processing efficiency, but the existing results mostly focus on single constellation signal processing.

On this basis, this paper makes a more in-depth study of the capture technology that is compatible with multiple constellation signal capabilities. A good compatible signal capture strategy can improve not only the acquiring performance of receiver but also the processing capabilities of weak GNSS signal and greatly reduce the consumption of resources. However, the difference of the signal system of each system brings some difficulties to the design of the compatible capture strategy. The compatible acquisition strategy based on GPU architecture proposed in this paper can not only capture GPS and other single constellation signals but also achieve compatible acquisition of four major system signals with different code lengths, code periods, or code rates by configuring system parameters. In addition, the design makes full use of the advantages of GPU parallel processing and improves the efficiency of signal acquisition. The experimental results show that the successful capture of the eight satellites takes only 2.312 s.

## 2 Methodology

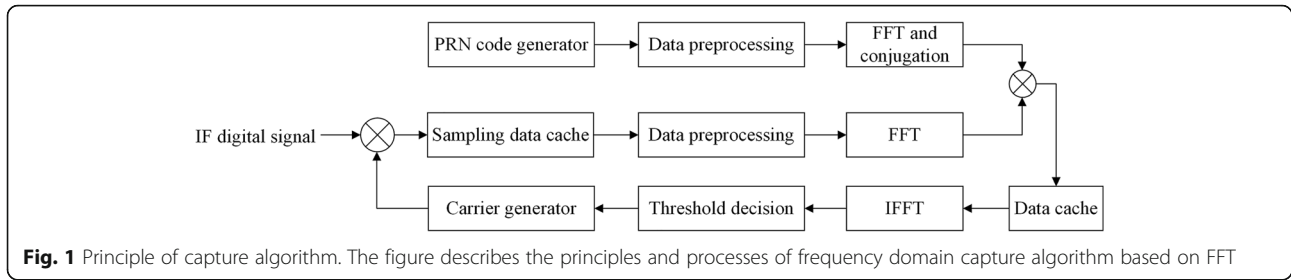
### 2.1 Analysis of acquisition algorithm

The purpose of signal acquisition is to determine the satellite that can be observed and to estimate the rough code phase and carrier Doppler shift [21, 22]. The essence of the capture is to use the strong auto-correlation of the pseudo random noise (PRN) code in the navigation signal to identify the navigation signal from the noise [23]. In this paper, the fast acquisition method based on fast Fourier transformation (FFT) is adopted. The core is to transform the time-domain correlation computation between the received signal and the local pseudo code into the multiplication calculation in frequency domain by using the circle correlation theorem, thus greatly reducing the acquisition time. The principle of the frequency domain capture algorithm based on FFT is shown in Fig. 1.

Usually, when the algorithm is used to capture the signal, the number of arithmetic points should be an integer power of 2. Therefore, it is necessary to process the data through a data pre-processing unit before the FFT operation. But this paper uses GPU high-performance computing platform, its internal cuFFT system kernel function library will automatically optimize and accelerate  $2^a \times 3^b \times 5^c \times 7^d$ . So there is no need to consider whether the length of the processed data meets the basic 2-FFT operation. Therefore, the data pre-processing operation designed in this paper is mainly used to improve the compatibility of GNSS signal acquisition, so that it can bring the data processing efficiency of the receiver and improve the performance and speed of the signal capture.

### 2.2 Principle of compatible acquisition

This section gives an analysis of the principle of GNSS signal compatibility acquisition based on GPU technology. Whether they capture GPS signals or other constellation signals, they all have the same capture and acquisition process, but there are also some differences caused by different signal characteristics [24]. In this paper, GPS L1 C/A, Beidou B1I, Galileo E1B, and GLONASS L1 signals are studied as examples. The difference between the four signals is shown in Table 1. It can be seen from the table that the differences of the four large system signals include the code length, the code rate, the modulation mode, and the way of the RPN code generation. Especially in the modulation mode, different from the traditional BPSK signal, the auto-correlation function of the BOC signal has many peaks, and it is necessary to pay attention to it. Therefore, in the design process of the compatible capture strategy, the similarities and differences of the system signal should be fully considered.



In this design, in order to make the system compatible with the signals with different characteristics, the common part is integration design into a public module, the differential part (PRN code generator module and data processing module) are designed into an independent sub module, and realizes the seamless connection between the common module and the difference between modules, so as to improve the system code reuse rate and data processing efficiency.

As shown in Fig. 1, the system contains two data pre-processing modules, one of which is data pre-processing for the output of the PRN code generator module and the other one is the pretreatment of the intermediate frequency (IF) sampling data. The former uses the method of up-sampling to achieve the expansion of the local PRN code, so that it can meet the needs of acquisition all of GNSS signal. According to the Nyquist sampling theorem, this paper define the bit wide of data as two times of 4.092 MHz, which means that the module will produce PRN code data with a frequency of 8.182 MHz. Therefore, during the process of capture, the length of the pseudo-code data is 8182 after the pre-processing of each input 1 ms. And the other pre-processing module mainly uses the methods of down-sampling and sub-sampling to reduce the sampling rate of the IF data, so that the width of input GNSS signal can meet the requirements of the compatible capture. In this paper, the short-time coherent integration algorithm is used to reduce the sampling data of the intermediate frequency of the GNSS signal. Usually, satellite data

contains a number of satellite signals, and one of the satellite signals is used for analysis. Suppose that the IF sampling data  $S(t)$  obtained in the satellite with PRN K and the result of the multiplication of the co-phase component and the quadrature component in the local carrier generator are seen as two intermediate variables  $S_I(t)$  and  $S_Q(t)$ , that is

$$S_I(t) = s(t) \cdot \sqrt{2} \cos[(\omega_{IF} + \Delta\omega)t] = D_k(nt_s)C_k(nt_s) \cos(\Delta\omega t) \quad (1)$$

$$S_Q(t) = s(t) \cdot \sqrt{2} \sin[(\omega_{IF} + \Delta\omega)t] = D_k(nt_s)C_k(nt_s) \sin(\Delta\omega t) \quad (2)$$

Here,  $t_s$  is the sampling period, showing  $N_{ts} = 1$  ms, where  $C_k$  is the PRN code,  $D_k$  is the bits of the navigation message, and  $\omega$  is the Doppler frequency. If the IF signal  $S(t)$  after ADC has the same frequency with the local carrier generator, then  $\omega$  is 0. When the short-time PRN code does not jump, the  $S_I(t)$  and the  $S_Q(t)$  values are constant, the strength of accumulated data block signal has been enhanced; when the frequency is not at the same time, the  $S_I(t)$  and  $S_Q(t)$  values of the existence of a Doppler frequency sine and cosine component of  $\omega$ , the cumulative signal strength reduce, which means not related to accumulated signal data in the block strength has been weakened. When  $\omega = 0$  and short-time PRN code jumping change takes place, such a situation only takes place for once in the complete 1 PRN code period and it will not cause an influence on the PRN code signal treatment in the entire period.

**Table 1** The difference between four major constellation systems

GNSS signal	PRN code length	PRN code period (ms)	modulation mode	PRN code type
GPS L1 C/A	1023	1	BPSK	Gold code
Galileo E1B	4092	4	BOC	Memory code
GLONASS L1	511	1	BPSK	M sequence
Beidou B1I	2046	1	BPSK	Gold code

The table shows the difference between the four constellations in terms of the code length, the code rate, the mode of modulation, and the way of code generation of the PRN

### 2.3 Design of compatible capture strategy based on GPU

Based on the above discussion, this article presents a compatible capture structure based on GPU, whose structure is shown in Fig. 2. Depending on the CPU platform, GNSS signal acquisition can be carried out in parallel by multithreading. Each thread can reuse the signal acquisition module, so that different data resources can be processed by using the same instructions.

As shown in the diagram, the system is mainly composed of channel parameter control setting module, related energy accumulation and signal acquisition module, pseudo-code generator, sampling data memory, and two data pre-processing modules. Among them, the first input first output (FIFO) and channel parameter control are used to store the data and control parameters for each channel, respectively. The multi-channel PRN generator is responsible for producing a local pseudo code of each system, and the pseudo code should be sent into the data pre-processing module to realize the bit expansion of the code data. The sampling data memory is used to store the 1-ms satellite signal to be captured, and the output data of this module also needs to be pre-processed. Two data pre-processing modules and a related energy accumulation and signal acquisition module are designed to realize the parallel treatment for GNSS signal capture, so as to achieve the features of multi-logic channel capture and multi-functional reuse. Because the official ICD document of the Galileo system does not give the way of its PRN code generation, the PRN code generation of Galileo signal can only be realized by register storage. Therefore, the system's pseudo-code generator module includes a universal code generator and a memory code controller. The former is used to generate PRN codes for GPS, BDS, and GLONASS systems, and the latter is used in the Galileo system.

The flow of the GNSS compatible acquisition algorithm is shown in Fig. 3. The capture module performs the signal capture operation of the GNSS based on the captured execution command information received by the system load monitoring module. When the capture

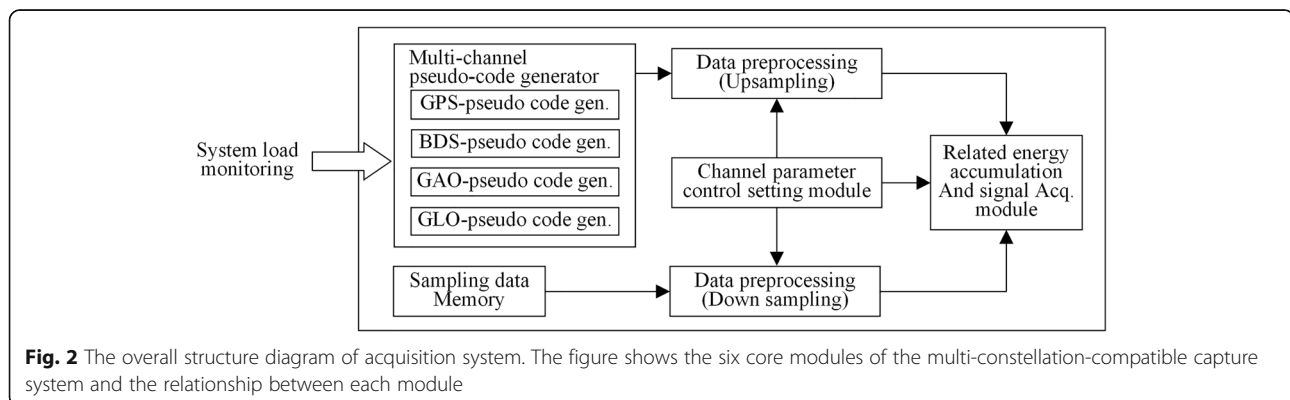
phase begins, the acquisition module acquires the state of signal acquisition from the interior of the GNSS receiver system. During the process of capturing, the receiver depending on the GPU high-performance processor concurrently opens the Kernel function of multiple cores, and each Kernel function performs PRN code acquisition in a frequency domain for a satellite. After the capture is finished, the system stores the capture results and transmits them to the tracking channels for subsequent signal processing. The system also makes full use of the flow processing mechanism of the GPU high-performance computing platform. In the process of acquisition processing, another thread can execute signal data from the CPU memory to the GPU memory copy operation, so as to make more efficient capture.

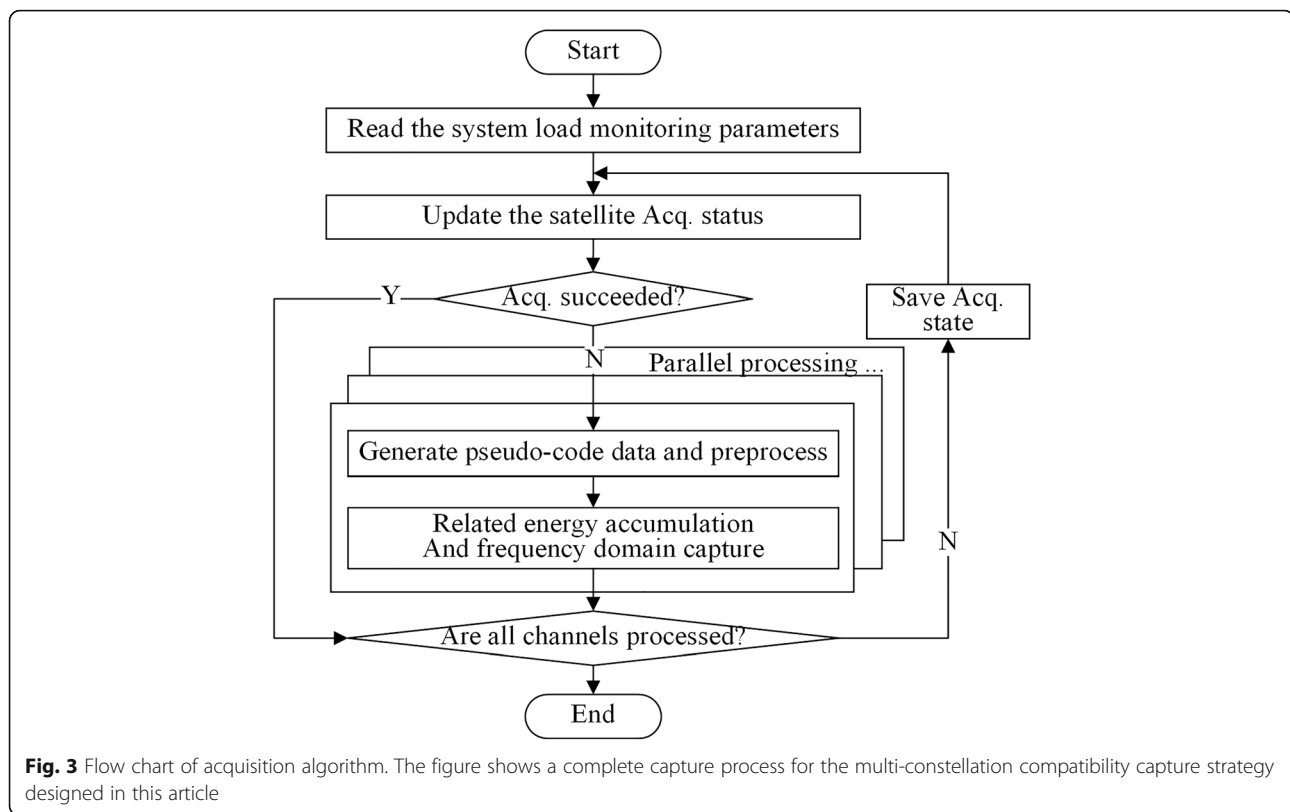
In this design, the related energy accumulation and signal acquisition algorithm modules are designed into separate thread safety structures, so as to achieve parallel acquisition of multiple signals without confusion. In addition, the structure can independently carry out coherent accumulation and incoherent accumulation of data point by point. The processing process of the correlation energy accumulation and frequency domain capture of a logical channel is shown in the Fig. 4.

### 2.4 Summary of system characteristics and compatibility

Based on the above design structure and processing flow, the design scheme proposed in this paper can implement the different system-compatible acquisition according to different satellite navigation signals. Once acquisition is achieved, a rough estimate value of the code phase and Doppler can be obtained. The compatible capture designed in this article has the following characteristics.

- Multi-channel parallel search: Based on the GPU platform, the GNSS signal capture is carried out in parallel by multithreading. And each thread reuses the signal capture module to perform the different dataprocessing with the same instruction.





- Support different signals with 1023 integer multiples of the code length: The up-sampling of the PRN code through the data pretreatment module which ensures the width of the local pseudo-code data meets the design requirements of wide energy accumulation and signal capture module, according to the Nyquist criteria set receiver capture width 8192, making it suitable for all the GNSS signal capture requirements. In addition, the length of the correlative integral time is set by the channel parameters, so that the system can support the acquisition of signals with different code periods.
- Compatible with BPSK and BOC modulation modes: The use of a single logical channel can support BPSK signal acquisition. If the two logic channel carrier frequencies are set to the two peak values of the BOC signal and the incoherent cumulative results of the two are added together, the BOC signal can be captured.
- Support different code rate and sampling rate signals: The two data pre-processing modules mentioned in the previous article are the key to the implementation of this part of the compatibility
- Support the fine parallel acquisition of carrier Doppler frequency: By increasing the number of the FFT transform of the signal after despreading, the finer Doppler frequency estimation can be obtained,

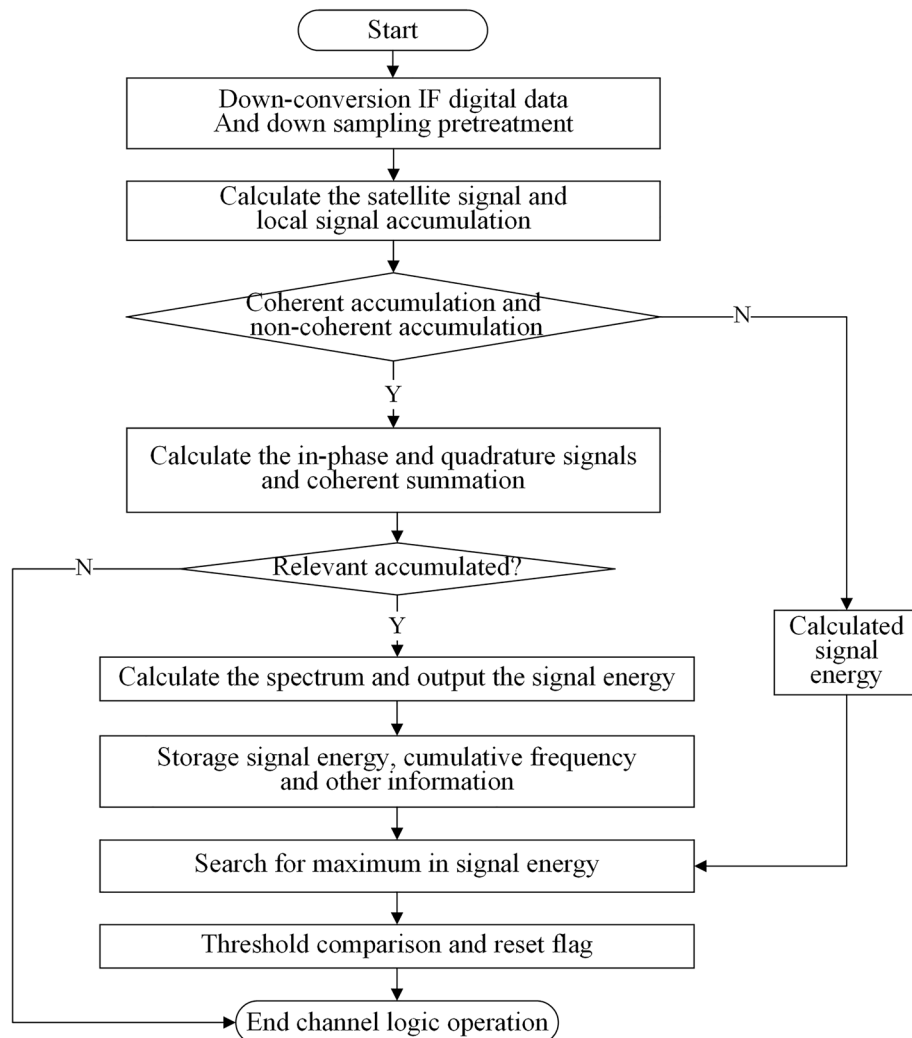
which is helpful for tracking a channel to track the signal more accurately.

- The number of code phase search is variable: By setting the channel parameters, the number of processing points and the number of code phase slips can be controlled, in order to achieve a variable number of code phase search.
- Accelerating capture by stream processing: Performance computing platform, when the GNSS signal capture operations, another thread will copy the GNSS signal data from the CPU memory to the GPU memory operation, thus causes the receiver to capture more quickly and efficiently. In addition, one or several satellite systems can be selectively captured by the system load monitor to improve capture efficiency.

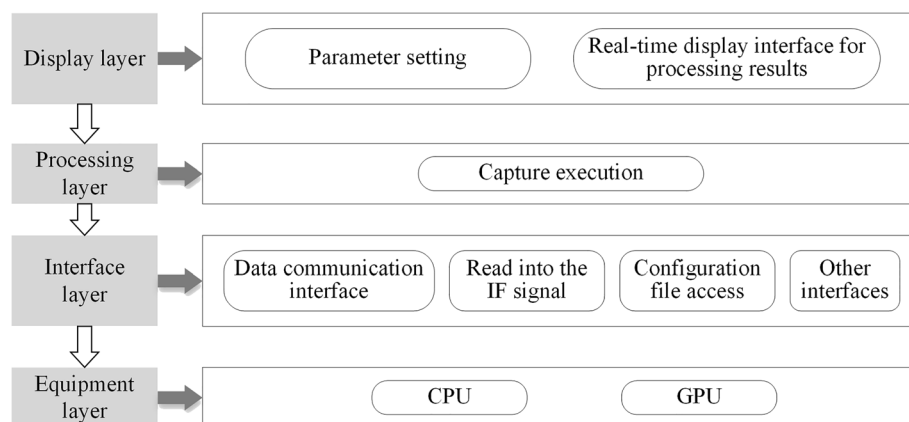
### 3 Results and discussion

#### 3.1 System prototype

The GNSS-compatible acquisition strategy proposed in this paper is based on the GPU high-performance computing platform, mainly built by NVIDIA GeForce GTX 850M graphics card and developed by compute unified device architecture (CUDA) integrated development kit on Visual Studio2010 platform. CUDA is a common parallel computing architecture launched by NVIDIA in



**Fig. 4** Procedure of logical channel. The figure shows the processing process of the correlation energy accumulation and frequency domain capture of a logical channel in the system



**Fig. 5** System prototype architecture diagram. The figure shows the system prototype of the compatible capture system designed in this article, which is mainly composed of the display layer, the processing layer, the interface layer, and the device layer



**Fig. 6** Acquisition parameter setting interface. This figure shows the interface of the system, where the user can configure the capture parameters through the interface before the capture starts

2006, which enables GPU to solve complex computing problems [25].

The system prototype designed based on GPU architecture proposed in this paper is mainly divided into the display layer, the processing layer, the interface layer, and the device layer, as shown in Fig. 5.

- Display layer: It provides the user with a simple system parameter configuration interface and present processing results.
- Processing layer: The processing requests submitted by the user are invoked efficiently through a good interface, and the capture processing of different signals can be executed according to the difference in the system settings. In addition, it makes full use of the advantages of GPU parallel operations to process signals and import programs in a modular way.
- Interface layer: It is used to specify the access methods of the corresponding devices and files, as well as the format of the data storage.
- Equipment layer: It is the lowest level of the whole system. The system is a high-performance computing platform based on GPU, where GPU is responsible for massive data high-performance parallel computing, and CPU is responsible for process control of program execution.
- In order to achieve the compatible capture of GNSS signals, the key and difficult point of this design is the signal capture part of the system processing layer.
- The specific development environment is shown as follows:
- Operating system: Windows10 x64 pro

- CPU: Intel(R) Core(TM) i7-4710MQ
- GPU: NVIDIA(R) GeForce(R) GTX 850M.
- CUDA version: Version 7.5.18

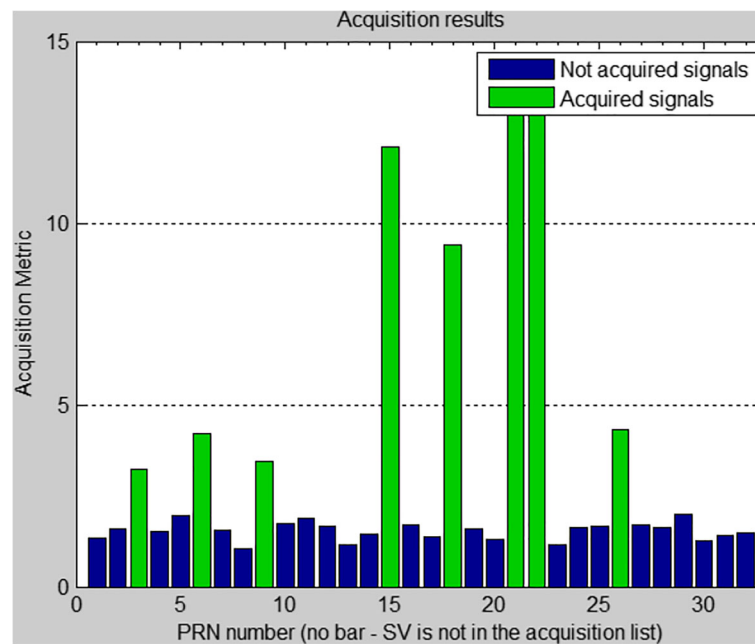
### 3.2 System implementation and test

Here, the capture structure based on GPU is tested, and the system's capture test interface is shown in Fig. 6. Before capturing, users must configure relevant information, including intermediate frequency, sampling rate, search bandwidth, and selection of satellite system. After setting up, signal acquisition operation can be carried out. Because of the limited hardware resources in the laboratory, the GPS digital IF signal is used as an example to test the system architecture designed in this paper. In order to highlight the high performance of GPU and the practical application ability of CPU to deal with complex

**Table 2** The satellite data captured by the GPU experimental platform

Satellite number	Ratio of peak values	Code phase (chips)	Carrier frequency(MHz)
21	13.56	13404	9.547480976105E + 06
22	12.66	6287	9.549720973969E + 06
15	10.47	36322	9.549975933075E + 06
18	9.36	20724	9.548300487517E + 06
6	4.03	28203	9.544348621368E + 06
9	3.42	4695	9.550895606995E + 06
3	3.39	34211	9.549948616028E + 06
26	3.07	26828	9.545067690276E + 06

The table shows the specific data of the satellite number, the code phase, the carrier frequency and the peak ratio of the 8 satellites captured in the experiment



**Fig. 7** The proportional peak diagram of satellite acquisition. This is an intuitive display of the results of the capture of 32 GPS satellites in the test. In the figure, the green bars represent successful acquisition and the blue bars represent a capture that failed

data, the test uses the large file data sampled by high sampling frequency. Therefore, the input signal is that the GPS satellite signal is sampled by the 38.192 MHz sampling frequency and the 4 bit sampling width of 50 s, and the frequency of the original signal is 9.548 MHz.

Here, the system successfully captured eight GPS satellites. The specific data of the satellite number, code phase, carrier frequency, and peak ratio of the eight satellites are shown as shown in Table 2. Figure 7 shows the visual effect of the capture results.

### 3.3 Analysis of system acquisition efficiency

In order to further verify the advantage of GPU for signal capture in efficiency, the capture efficiency of GPU and CPU is compared. In the same experimental environment, 20 groups of independent acquisition experiments based on CPU and GPU were completed respectively by using the same data. The average execution time of the two experimental groups to capture a single satellite and capture eight satellites was calculated. Among them, the execution time of GPU is determined by the special timing API provided by NAVIDA, and the running time of CPU is determined by the timer function provided by MFC. The comparison results are shown in Table 3.

The test results show that when the single satellite is captured, the signal capture base on GPU needs more execution time than the signal capture base on CPU.

This is because the initialization of the CUDA computing and the memory allocation will take a certain amount of time. And using GPU-based capture structure to traverse all satellites takes only 2.312 s time, which is nearly five times faster than that of all satellites captured by CPU. In fact, the highest frequency of i7-4710MQ CPU is up to 3.5 GHz, but its execution speed is still less than that of the GTX850M card with the highest 2.5 G frequency. It can be seen that the advantages of parallel processing of GPU are remarkable.

The following is a further analysis of the efficiency of this design using the GPU architecture for signal capture. Figure 8 is the percentage of the time consumed by each kernel when it is captured under this structure.

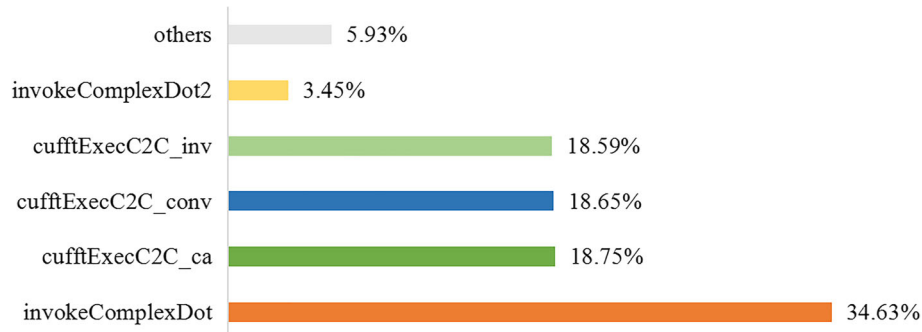
From the graph, it can be seen that the time spent in capturing the structure in this paper is mainly concentrated in the FFT computation performed by the three time cufftExecC2C kernel functions, accounting for 55.

**Table 3** Time consuming comparison between GPU and CPU

The number of satellites	Experimental mode	Execution time
1	Single satellite capture(CPU)	0.478 s
	Single satellite capture(GPU)	1.267 s
8	All satellite capture(CPU)	10.398 s
	All satellite capture(GPU)	2.312 s

The table shows the average time consumed contrast of the GPU and CPU capture satellites under the same experimental conditions





**Fig. 8** Time consumed percentages of each kernel. The figure is the time-occupying time consumed by each kernel in the execution of the capture process by the system designed in this article. Among them, the time of three FFT operations is up to 55.99%

99% of the total computation ratio (three times and FFT time). This part is the key to CPU's acceleration over CPU's execution of signal capture operations.

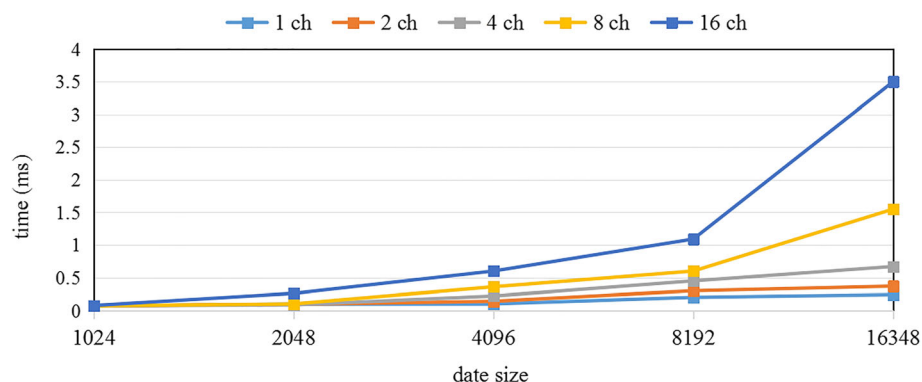
In addition, the number of points in the FFT operation designed by this article is variable, and the number of execution capture channels can also be set on its own. As the number of FFT point's increases and the number of parallel acquisition channels increases, the advantage of GPU as a high-performance parallel computing platform will become more and more obvious. Of course, this acceleration is not necessarily established with the increase in the amount of data. The time consumed of a FFT operation in this paper is changed with the number of channels and the number of points, as shown in Fig. 9.

To sum up, the new compatible capture structure based on GPU is proposed in this paper, which has high computing efficiency and very considerable time compression ratio.

#### 4 Conclusions

This paper is based on the research of CPU-based multi-constellation GNSS signal acquisition technology. Compatible capture is the key technology for

developing GNSS receiver. The compatible acquisition scheme proposed in this paper not only expands the format of captured signals, which almost supports all existing satellite system signals, but also achieves a parallel processing of GNSS signal data by using GPU, which improves the efficiency of capture processing. Using this structure to capture eight parallel satellites only takes 2.312 s, which is about five times faster than the CPU in the same case. In addition, the GPU high-performance computing platform used in this paper is built mainly by the NVIDIA GeForce GTX 850M graphics card and is implemented in conjunction with the CUDA programming environment. It is very convenient for ordinary computers to build the GPU high-performance computing platform which is proposed in this paper. It only needs to access a NVIDIA graphics card supporting a CUDA programming. Therefore, the design of this paper has good implementation and portability. In another words, making full use of the advantages of GPU and applying it to the processing of GNSS satellite signals is a new direction for the design of compatible receivers. Moreover, the design idea of this paper is redesigning



**Fig. 9** Time consumed change diagram of FFT operation. The figure shows the time consumed changes in the FFT operation as the number of channels and FFT points changes in the system

the GNSS system, which can provide a reference for the new research of how to allocate resources rationally and build an ideal multi-country shared civil GNSS and the next generation of satellite navigation system.

#### Abbreviations

ARM: Advanced RISC machines; CPU: Central processing unit; CUDA: Compute unified device architecture; DSP: Digital signal processing; FFT: Fast Fourier transformation; FIFO: First input first output; FPGA: Field programmable gate array; GNSS: Global navigation satellite system; GPU: Graphics processing unit; IF: Intermediate frequency; PRN: Pseudo random noise code

#### Acknowledgements

The authors would like to thank Prof. Wei Guo at the National Key Laboratory of Science and Technology on Communications of UESTC for the help, and Prof. Long Jin and Prof. Yonglun Luo at the Research Institute of Electronic Science and Technology of UESTC for their assistance in the GPU and DSP. The authors also want to thank the Research Institute of Electronic Science and Technology and Key Laboratory of Integrated Electronic System, Ministry of Education for their support to this research. However, the opinions expressed in this paper are solely those of the authors.

#### Authors' contributions

CG is the main writer of this paper. He put forward the main idea, completed the experimental test, and analyzed the result. The overall design scheme of GPU-based compatible acquisition algorithm was mainly proposed by CG, and BX assisted him in completing the design. ZT gave some important suggestions and help for the technology of GNSS signal processing. All authors read and approved the final manuscript.

#### Competing interests

The authors declare that they have no competing interests.

#### Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

#### Author details

<sup>1</sup>National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China.

<sup>2</sup>Research Institute of Electronic Science and Technology, University of Electronic Science and Technology of China, Chengdu, China.

Received: 1 February 2018 Accepted: 26 April 2018

Published online: 09 May 2018

#### References

- Zhou, Y Li, J Liu, et al., Equality constrained robust measurement fusion for adaptive Kalman filter based heterogeneous multi-sensor navigation. *IEEE Trans. Aerosp. Electron. Syst.* **49**(4), 2146–2157 (2013)
- J Leclerc, C Botteron, PA Farine, Comparison framework of FPGA-based GNSS signals acquisition architectures. *IEEE Trans. Aerosp. Electron. Syst.* **49**(3), 1497–1518 (2013)
- Fortin M, Bourdeau F, Landry R, Implementation strategies for a software-compensated FFT-based generic acquisition architecture with minimal FPGA resources. *Navigation - J Institute Navigation.* **62**(3), 71–188 (2015).
- J Leclerc, C Botteron, PA Farine, Acquisition of modern GNSS signals using a modified parallel code-phase search architecture. *Signal Process.* **95**(2), 177–191 (2014)
- Dovis F, Mulassano P, Gramazio A, SDR technology applied to Galileo receivers. *Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS '02)*, 2002.
- S Khan, A Borsic, P Manwaring, et al., FPGA based high speed data acquisition system for electrical impedance tomography. *J Phys Conf Se.* **434**(1), 012081 (2013)
- Xiaolei, Yongrong, Jianye, et al., Design and realization of synchronization circuit for GPS software receiver based on FPGA. *J. Syst. Eng. Electron.* **21**(1), 20–26 (2010)
- Z Zhou, Y Li, J Zhang, et al., Integrated navigation system for a low-cost quadrotor aerial vehicle in the presence of rotor influences. *J. Surv. Eng.* **143**(1), 05016006 (2016)
- H Wang, J Chang, J Lv, et al., An implementation scheme of GNSS signal simulator based on DSP and FPGA. *International Conference on Computer Science and Service System, IEEE* (2011), pp. 27–29
- V Chakravarthy, J Tsui, D Lin, et al., Software GPS receiver. *GPS Solut* **5**(2), 63–70 (2001)
- T Jokitalo, K Kaisti, V Karttunen, et al., A CPU-friendly approach to on-demand positioning with a software GNSS receiver. *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), IEEE* (2010), pp. 14–15
- H Shamoto, K Shirahata, A Drozd, et al., GPU-accelerated large-scale distributed sorting coping with device memory capacity. *IEEE Transactions on Big Data* **2**(1), 57–69 (2016)
- J Wu, Z Song, G Jeon, GPU-parallel implementation of the edge-directed adaptive intra-field deinterlacing method. *J. Disp. Technol.* **10**(9), 746–753 (2017)
- Lobeiras J, Amor M, Doallo R, Designing efficient index-digit algorithms for CUDA GPU architectures. *IEEE Transactions on Parallel & Distributed Systems*, **27**(5), 1331–1343 (2016).
- M Garland, SL Grand, J Nickolls, et al., Parallel computing experiences with CUDA. *Micro IEEE* **28**(4), 13–27 (2008)
- Z Yu, L Eeckhout, N Goswami, et al., GPGPU-MiniBench: accelerating GPGPU micro-architecture simulation. *IEEE Trans. Comput.* **64**(11), 3153–3166 (2015)
- S Ando, F Ino, T Fujiwara, et al., Enumerating joint weight of a binary linear code using parallel architectures: multi-core CPUs and GPUs. *IJNC* **5**(2), 290–303 (2015)
- SH Im, GI Jee, Software-based real-time GNSS signal generation and processing using a graphic processing unit (GPU). *J Positioning Navigation Timing* **3**(3), 99–105 (2014)
- L Xu, NI Ziedan, X Niu, et al., Correlation acceleration in GNSS software receivers using a CUDA-enabled GPU. *GPS Solutions* **21**(1), 1–12 (2017)
- T Hobiger, T Gotoh, J Amagai, et al., A GPU based real-time GPS software receiver. *GPS Solutions* **14**(2), 207–216 (2010)
- Y Wang, G Mao, Fast GNSS satellite signal acquisition method based on multiple resamplings. *Eurasip J Adv Signal Processing* **2016**(1), 109 (2016)
- BM Ledvina, ML Psiaki, SP Powell, et al., Bit-wise parallel algorithms for efficient software correlation applied to a GPS software receiver. *IEEE Trans. Wirel. Commun.* **3**(5), 1469–1473 (2004)
- D Akopian, Fast FFT based GPS satellite acquisition methods. *Radar, Sonar and Navigation, IEE Proceedings* **152**(4), 277–286 (2005)
- L Huang, Acquisition algorithm for GNSS multi-constellation signals. *Sci. Sinica* **41**(5), 226–232 (2011)
- E Lindholm, J Nickolls, S Oberman, et al., NVIDIA tesla: a unified graphics and computing architecture. *IEEE Micro* **28**(2), 39–55 (2008)

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)