Verification of GPS Remote Forwarding Experimental Platform

Junzheng Wang, Xiaoyi Feng(✉), and Deyue Zou

Dalian University of Technology, Linggong Street. 2, Dalian 116024, China
18041534683@163.com, zoudeyue@dlut.edu.cn

Abstract. With the rapid development of Global Positioning System (GPS) technology and its wide application, the performance and flexibility of GPS software receiver are increasingly expected. Existing receivers either are portable and taken outdoors as a whole, but unable to adjust the top-level algorithms in real-time; or can only simulate or observe fixed positions indoors, but unable to perform real-time multi-point measurements. Both have become unsuitable for the rapid development requirements of receivers. This paper introduces a GPS remote forwarding experimental platform, which configures the acquisition and tracking function at the transmitting end and the positioning solution function at the receiving end, respectively completed by two PC. Between PCS, based on TCP Socket communication, the two PCS are connected through the wireless AP bridge, and the received tracking data is transmitted to the MATLAB platform for positioning solution. The experimental platform described in this paper can achieve portability at the front end with good mobility, and the computing platform at the receiving end in the laboratory can be complex and flexible to accommodate diverse algorithm development.

Keywords: GPS · Acquisition · Tracking · Navigation · Forwarding

1 Introduction

The application scope of GPS satellite navigation technology is expanding. GPS acquisition, tracking, and navigation are essential components of modern navigation systems. In paper 1, Wu et al. discuss the use of software GPS receivers for signal acquisition and tracking. In paper 2, Lita et al. focus on developing new equipment and algorithms for extracting navigation data bits from raw GPS signals, especially in challenging environments. In paper 3, Hyun et al. delve into the development of acquisition and tracking schemes for a GPS L5 receiver, utilizing FFT algorithms for cold start acquisition. In paper 4, Lei et al. present an acquisition algorithm based on FFT for GPS signal acquisition, emphasizing the importance of detecting satellite signals for tracking and decoding navigation information. In paper 5, Langer et al. introduce non-coherent Deeply Coupled GPS/INS integration in pedestrian navigation systems to enhance position accuracy in weak signal conditions. In paper 6, Jwo et al. propose a sensor fusion method for Ultra-Tightly Coupled GPS/INS integrated navigation, comparing various nonlinear filtering

H.-H. Chen and W. Meng (Eds.): WiSATS 2024, LNICST 606, pp. 399–408, 2025.
https://doi.org/10.1007/978-3-031-86203-8_31

approaches for optimal performance. In paper 7, Vu et al. implement the Vector Tracking Loop algorithm in modern GPS receivers to calculate position and velocity using Kalman filters. In paper 8, Capuano explores the feasibility of using GNSS for autonomous navigation in lunar missions, highlighting the design requirements for a code-based GNSS receiver. In paper 9, Phyo et al. focus on the implementation and analysis of signal tracking loops for software-defined GPS receivers, specifically carrier and code tracking loops on GPS L1 signals. Lastly, In paper 10, Savas et al. conduct a comparative performance analysis of GPS L1 C/A and GPS L5 acquisition and tracking methods under polar and equatorial scintillations. They assess the robustness of different acquisition methods in terms of detection probabilities, peak-to-noise floor ratios, and execution time, particularly in challenging propagation environments. These studies collectively contribute to the advancement of GPS acquisition, tracking, and navigation technologies. However, in some environments, such as dense high-rise areas, mountainous areas, tunnels and other complex terrain areas, GPS signal is blocked or interfered seriously, there is signal attenuation and multipath effect interference, there is a large error or can not be located. Among the existing GPS software receivers, one is portable and the other is indoor. The former can be applied outdoors as a whole, but it is not possible to adjust the front-end algorithm in real time, and the latter is completely indoor to simulate or observe a fixed position and cannot perform real-time multi-point measurements. Both of the above are not suitable for the rapid development of receivers. Therefore, this paper designs a GPS remote experimental platform, which solves the limitations of traditional software receiver, makes the front end portable, transmits data back to the indoor PC end, and realizes real-time parameter change. The feasibility of the GPS remote forwarding platform under different environmental conditions is verified.

2 System Design

2.1 Overall Architecture Design

A satellite navigation signal software-defined radio (SDR) receiver is a receiver based on a computer platform that acquires, processes, and localizes satellite signals through software. Compared to hardware receivers, software receivers offer greater flexibility, adaptability to different satellite signals, and are more cost-effective. Against the backdrop of the content studied in this paper, a software receiver is chosen as the form of the receiver for the GPS experimental platform. In satellite navigation systems, personal computers (PCs) and field-programmable gate arrays (FPGAs) are two commonly used baseband processing platforms. PCs are capable of handling complex algorithms and large volumes of data, meeting the high-performance requirements of satellite navigation signal baseband processing. Moreover, PC programming is flexible and convenient, making it easy to implement various algorithms and functions. The software development resources are abundant, facilitating the utilization of existing reference routines and tools for development. Therefore, the GPS experimental platform in this study adopts a PC as the baseband processing platform.

In this paper, MATLAB is selected as the software programming development environment for the GPS remote forwarding experimental platform. Building upon the existing mature GPS simulation signal software sources and MATLAB algorithms for signal acquisition, tracking, and position solution, a MATLAB program is developed to enable the modification of configuration file parameters through a graphical user interface (GUI). The program then automatically generates tracking results for the simulated satellite signals and stores them in mat files within the corresponding folder. Subsequently, the tracking results are transmitted to the wireless transmission receiving end via a wireless transparent transmission device, ultimately achieving position solution calculations.

The system primarily comprises the GPS software component and the wireless transparent transmission hardware component, with the software-defined radio (SDR) receiver serving as the core component responsible for acquiring and processing data from the GPS software-based signal source. Additionally, the system integrates MATLAB programs, offering users advanced data processing and analysis capabilities. The system supports the execution of exe files, facilitating more convenient operation. Through wireless means, the system is capable of transparently transmitting tracking data for subsequent position solution calculations. Figure 1 illustrates the overall architecture of the GPS remote forwarding experimental platform program and its functional modules.

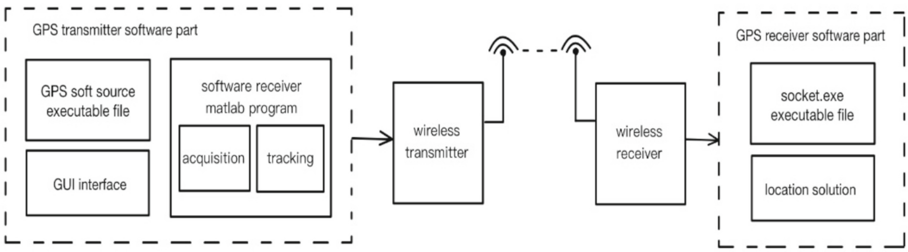

Fig. 1. Overall Architecture of the GPS Remote Forwarding Experimental Platform Program.

Functionally, the software system is capable of acquiring and tracking GPS signals and performing position solution calculations on these signals, which is essential for ensuring precise positioning and navigation at the receiving terminal. Operations for configuring the GPS software-based source parameters can be interacted with through a user-friendly GUI, allowing users to easily configure and control the software system. In summary, the GPS remote forwarding experimental platform, with its comprehensive functionality and user-friendly GUI, offers robust technical support for GNSS applications.

2.2 Transmitter Program Design

The software component of the transmitter for the GPS remote experimental platform primarily consists of a series of MATLAB programs and an exe program for the simulation signal software source. The MATLAB series programs include several functional sub-programs, such as the main program, GUI interface program, parameter storage program, acquisition and tracking program, plotting program, and parameter writing program. The simulation signal software source exe program is a pre-packaged executable that can be invoked to generate simulated satellite signals. To call this program, the basic configuration and trajectory configuration contents must be stored in two txt parameter files in advance. This allows the simulation signal software source to produce corresponding analog simulation satellite signals based on different parameter configurations. The author has modified the software source exe to read parameter configuration documents from a fixed address and re-packaged it as an executable exe file, enabling MATLAB to invoke the software source with a single command, which greatly facilitates the generation of simulated signals.

To avoid the cumbersome process of manually entering the configuration file addresses in sequence to invoke each program due to the GPS simulation software source and the acquisition and tracking program being separate entities, and also to facilitate human-machine interaction and the modification and storage of simulation parameters, the author has developed a UI tuning control interface based on the MATLAB environment, utilizing GUIDE. This interface enables visual, convenient modification and storage of parameters related to basic configuration and trajectory configuration, allowing the simulation tracking results to dynamically change with parameter adjustments. The GUI interface is depicted in Fig. 2.

Fig. 2. The GUI Interface.

Specifically, the program invocation process involves the following steps: initially, the main program is executed, which directly calls the GUI interface program. After the user enters the parameters into the corresponding fields in the GUI interface, they click the

"One-Key Generation" button on the GUI interface. The callback function of this button's purpose is to save the user-entered configuration parameters to two parameter documents. After the parameters are stored, the control returns to the main program, which continues with the subsequent procedures. Subsequently, within MATLAB, the GPS simulation software source is invoked, generating the corresponding GPS simulation signals based on the parameters input by the user through the GUI. Next, in MATLAB, the GPS simulation signals are captured and tracked. Finally, the tracking results are generated. These tracking results will later be transmitted from the transmitting end PC to the receiving end PC using a wireless bridge, preparing the necessary data for subsequent positioning solution calculations at the receiving end PC. The system module flowchart is depicted in Fig. 3.

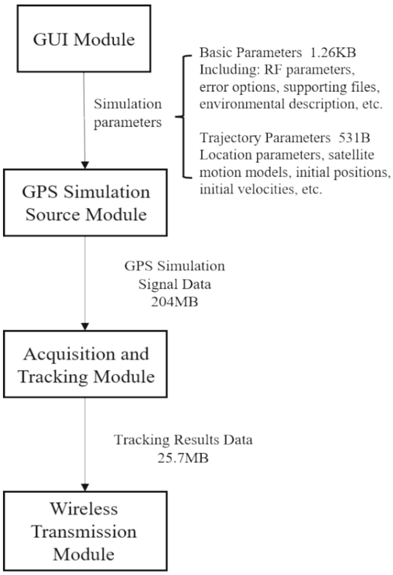

Fig. 3. System Module Flowchart.

2.3 Receiver Program Design

This design also implements a one-key operation, and the program invocation flow is depicted in Fig. 4.

In this system, the wireless receiver module transmits the tracking results to the navigation module, where the information included in the tracking results is shown in Table 1.

The navigation function is mainly composed of the following eight parts:

(1) Evaluate the Conformance to Position Calculation Criteria.
(2) Locate the Starting Point of the Frame Header Position.
(3) Decode the Ephemeris Data.

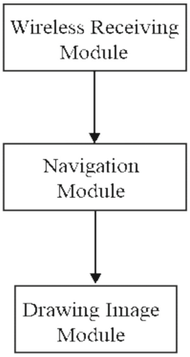

Fig. 4. Receiver Program Flowchart.

Table 1. The information included in the tracking results.

Parameters	Information
absoluteSample	Absolute sample position of C/A code in the record
codeFreq	Frequency of C/A code
carrFreq	Frequency of the carrier being tracked
I_P, I_E, I_L	Real part signals of prompt, early and late channels
Q_P, Q_E, Q_L	Imaginary part signals of prompt, early and late channels
dllDiscr	The phase difference between the code input signal and the copy signal
dllDiscrFilt	The offset of the code NCO
pllDiscr	Phase difference between the carrier input signal and the replicated signal
pllDiscrFilt	The offset of the carrier NCO
PRN	Pseudo-random noise

(4) Determine if the Number of Detected Satellites Remains Above Three.
(5) Utilize the Least Squares Method to Calculate Pseudorange, to Fit the Difference Between the Observed Arrival Time of the Satellite Signal at the Receiver and the Predicted Arrival Time.
(6) Determine the Satellite Position and Complete Clock Correction.
(7) Calculate the Receiver Position.
(8) Convert from Earth-Centered, Earth-Fixed (ECEF) Coordinate System to Geodetic Coordinate System and UTM Coordinate System.

Finally, the measured pseudo code and receiver coordinates were passed to the drawing module to draw the coordinate location map.

2.4 Wireless Transparent Transmission Design

In this design, a wireless transparent transmission module is selected, with the Mikrotik Metal 52 ac RBMetalG-52SHPacn dual-band wireless AP bridge used as the wireless transmission and reception endpoint. This type of bridge is configured using the Winbox software platform. During the initialization configuration of the bridge, it is necessary to connect the transmit and receive bridge endpoints to a PC separately for configuration. Once configured, connecting both the transmit and receive bridges to power will enable them to connect automatically.

During the configuration process, the transmit-side bridge is set to server mode, while the receive-side bridge is set to client mode. After setting the bandwidth, channel, etc., step by step, the transmit and receive bridges can successfully establish a connection. Connecting both bridges to power and the respective PCs, if the bridge signal lights are all illuminated, it indicates a correct and successful connection. The bridge connectivity status diagram is shown in Fig. 5. It is important to note that for two PCs to communicate over Ethernet through the bridge, their IP addresses must be manually set to be in the same network segment as the transmit and receive bridge IPs, i.e., the first three segments of the IP addresses should be the same. Under the TCP protocol, the average information rate at the transmit end is around 145 Mbps, and at the receive end, it is around 152 Mbps. It can be considered that the overall information rate reaches approximately 150 Mbps, which satisfies the platform's required rate of 16 Mbps. Therefore, the connection is successful and meets the speed standards.

Fig. 5. Bridge Connectivity Status.

Socket communication is a crucial mechanism in network programming that enables data exchange between different computers over a network. In this design, TCP protocol is employed for transmission. A wireless network is established through the configuration of a wireless bridge, and all PCs are connected to this wireless network. Within this wireless network, file data received via socket can be parsed and processed, with the transmitted file being encapsulated into a socket. The various components of information are then separated and input into the MATLAB software for subsequent operations.

After successfully using the bridge to connect two PCs into a small local area network, open the TCP transmission assistant software, input the IP address of the receiving computer, and select the file to be transmitted in the software. In this experiment, the tracking results file generated by the transmitter of the GPS remote experiment platform is chosen for transmission. The size of this file varies with the parameters set in the GUI for each experiment and is approximately 20 MB. Once the file is selected, the wireless transparent transmission can be initiated.

3 Experimental Results and Analysis

To further test the communication range and penetration capability of the wireless transparent transmission module, three sets of experiments were conducted with two PCs at varying distances. The experiment scenarios were chosen in a more complex indoor environment to test the penetration ability and communication range. Four transmission scenarios were selected: short-range line-of-sight transmission, transmission across one wall, transmission across two walls, and transmission across three walls. The experimental setup required two PCs, a transmit and receive bridge, power cables, and Ethernet cables. One PC served as the sender, generating and transmitting tracking results, while the other PC was responsible for connecting to the receive bridge to receive the tracking results and perform subsequent position solution calculations. A schematic diagram of the experimental equipment setup is shown in Fig. 6.

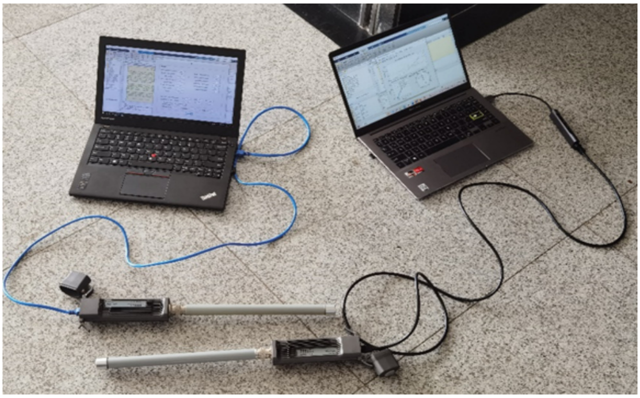

Fig. 6. Schematic Diagram of Experimental Equipment Connection.

The results of the four sets of testing experiments are presented in Table 2. The experiment was conducted based on four indicator dimensions: the interval of transparent transmission, the signal strength of the bridge, the time required for transparent transmission, and the positioning error.

The experimental results show that when the bridge signal strength is high (level 4) and there are no obstacles, the transparent transmission time is 10 s and the transmission is successful. However, when the transmission interval is a wall, the signal strength decreases to level 3, and the transmission time increases to 15 s, but the transmission is still successful. This result indicates that the transparent transmission performance is not significantly affected when the visual distance is 10 m or when there is an interval of one wall, with the performance being similar in both cases. When there is an interval of two walls, the signal strength decreases to level 2, and although the transmission

Table 2. Transparent Transmission Test Experimental Results.

Transmission interval	Signal intensity	Transmission time	Positioning error(m)
10 Meters Line of Sight	Level 4 (Maximum)	10 s	16.29441
One Wall	Level 3	15 s	22.88484
Two Walls	Level 2	22 s	18.59357
Three Walls	NaN	NaN	NaN

is successful, the required time increases to 22 s. This indicates that obstacles have a significant impact on the transparent transmission performance.

However, when the number of obstacles increases to three walls, even with the same transmission power, the transmission fails, with the signal strength being 0 and the bridge being unable to search for and connect with each other. This further highlights the limiting effect of obstacles on the transparent transmission performance.

In terms of positioning solution, as long as the transmission is guaranteed, the solution can be obtained in real-time. The error in longitude and latitude, converted to meters, shows that the error is stable within 30 m. Moreover, as the error does not increase with the thickness of the wall, through comparative experiments (transmission without walls), it is concluded that these errors are systematic errors. This means that, due to the reliable and trustworthy nature of TCP transmission, the positioning solution error is only a stable systematic error within 30 m as long as the transmission can proceed smoothly.

In summary, the experimental results indicate that the transmission interval has a significant impact on the bridge signal strength and transparent transmission performance. In practical applications, by considering the specific scenarios and requirements, and taking into account these factors comprehensively, it can be concluded that the GPS experimental platform can fully meet the requirements of the actual scenario with a transmission distance of 50 to 100 m without obstacles and within the visual range. It possesses the transparent transmission performance required in practical applications.

4 Conclusion

This paper investigates the scenario of indoor reception of navigation and positioning signals and constructs a GPS remote forwarding experimental platform to implement wireless remote signal forwarding and processing. When the experimental environment changes, in the case of wall barrier, the transmission time will increase with the increase of wall barrier, and the transmission reaches the limit when the thickness of three walls is increased. In the case of open and unobstructed, the transmission time will increase with the increase of distance, and the transmission can be stable at the farthest 100 m. In the above cases, and in the case of ensuring feasible transmission, the solution results are in line with the standard, which is consistent with the solution results when the environment has not changed. This paper provides a new verification idea and experimental basis for the development of future GPS remote experimental platform.

Acknowledgements. Thanks to Prof. Han Shuai and Prof. Meng Weixiao of Harbin Institute of Technology for providing software support for this study.

References

1. Wu, F., Zhang, K., Yasuda, A.: GPS signal acquisition and tracking using software GPS receiver. In: Proceedings of the Eighth International Symposium on Signal Processing and Its Applications (2005)
2. Lita, I., Visan, D.A., Popa, I.: Localization system based on enhanced software GPS receiver. In: 2006 29th International Spring Seminar on Electronics Technology, St. Marienthal, Germany (2006)
3. Hyun, C.S., Hyun, K.J., Shin, C.S., Lee, S.U., Kim, J.H.: Acquisition and tracking schemes for A GPS L5 receiver. In: 2008 International Conference on Control, Automation and Systems, Seoul (2008)
4. Lei, Q., Lei, L.: GPS signal acquisition based on FFT. In: 2010 Second International Conference on Information Technology and Computer Science, Kiev, Ukraine (2010)
5. Langer, M., Kiesel, S., Ascher, C., Trommer, G.F.: Deeply coupled GPS/INS integration in pedestrian navigation systems in weak signal conditions. In: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, NSW, Australia (2012)
6. Jwo, D.-J., Yang, C.-F., Chuang, C.-H., Lin, K.-C.: A Novel design for the ultra-tightly coupled GPS/INS navigation system. J. Navig. **65**(4), 717–747 (2016)
7. Vu, B.N., Andrle, M.: Implementation of vector tracking loop algorithm in modern GPS receiver. In: 2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC), Colorado Springs, CO, USA (2014)
8. Capuano, V.: GNSS-based navigation for lunar missions. In: Proceedings of the 27th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2014), Tampa, Florida (2016)
9. Phyo, A.S.S., Tun, H.M., Mon, A. and Pha, S.H.: Implementation and Analysis of Signal Tracking Loops for Software Defined GPS Receiver, In: Communications (2020)
10. Savas, C., Falco, G., Dovis, F.: A Comparative Performance Analysis of GPS L1 C/A, L5 Acquisition and Tracking Stages Under Polar and Equatorial Scintillations, In: IEEE Transactions on Aerospace and Electronic Systems, vol. 57, no. 1, pp. 227–244 (2021)