Space Science Mission Operation System Joint Test Method Research

Haiyan Wu [1]
Center for Space Science and Applied Research, Chinese Academy of Science, Beijing, China, 100190
Graduate University, Chinese Academy of Science, Beijing, China, 100049

Xin Meng.[2]
Center for Space Science and Applied Research, Chinese Academy of Science, Beijing, China, 100190

Yuzhu Zhang[3]
Center for Space Science and Applied Research, Chinese Academy of Science, Beijing, China, 100190
Graduate University, Chinese Academy of Science, Beijing, China, 100049

Zimin Zou.[4], Jizhou Tong.[5], Xiaodong.[6], Dalin Li.[7]
Center for Space Science and Applied Research, Chinese Academy of Science, Beijing, China, 100190

Testing is important for Space Science Satellite Ground Application System (SSSGAS) development, which usually takes a lot of time. This paper introduces a System Joint Test Platform that could be applied to SSSGAS's development, testing and verification. SJTP's purpose is to check whether SSSGAS's various subsystems and functional module interface is in line with the system interface specification so to reduce the workload of system testing. SJTP can simulate SSSGAS's various subsystems or functional module interface functions and processes in order to do debugging with the measured subsystem / function module interface; it can also simulate and test the system's process by connecting with one or more of the actual subsystem / functional modules together to emulate the complete SSSGAS.

Nomenclature

SJTP	=	System Joint Test Platform
SSSGAS	=	Space Science Satellite Ground Application System
SSMOS	=	Space Science Mission Operation Subsystem
SSDS	=	Space Science Data Subsystem
SSAS	=	Space Science Application Subsystem
DRS	=	Data Receiving Subsystem
TC&C	=	Telecommand and control

I. Introduction

JOINT test has always been critical during the production process of large, complex systems[1], especially the Space Science Satellite Ground Application System (SSSGAS). In the past, due to research units' different geographic locations, the joint test required all research units to gather which is a waste of time, cost and manpower. Therefore, by designing a generic testing tool, and use it to do pre-integration test at each research units. Then, after this period,

[1] Doctoral Candidate, Space Information & Simulation Laboratory, 8701 Beijing, Not an AIAA Member.
[2] Professor, Space Information & Simulation Laboratory, 8701 Beijing, Not an AIAA Member.
[3] Doctoral Candidate, Space Information & Simulation Laboratory, 8701 Beijing, Not an AIAA Member.
[4] Professor, Data and Net Center, 8701 Beijing, Not an AIAA Member.
[5] Engineer, Data and Net Center, 8701 Beijing, Not an AIAA Member.
[6] Engineer, Space Information & Simulation Laboratory, 8701 Beijing, Not an AIAA Member.
[7] Engineer, Space Information & Simulation Laboratory, 8701 Beijing, Not an AIAA Member.

these units could be gathered for the integration test. In this way, time and manpower could be saved, thus improving work efficiency.

At present, simple test systems are developed and used in some satellite ground application systems' development process. For example, a remote sensing satellite system joint test system[2] whose main idea is the unified management of all interface files; it checks sub-systems' the interface files before the test and displays the files received, then replies manually; but this system does not focus on system workflows, instead, it mainly dependents on the operator to check the flow and file contents constraints, state transfer control and function implementation.

In this paper, a method is proposed to implement testing process automation, which provides flexible configuration and management capabilities. By simulating external interfaces and some features of the SSSGAS, the joint test could be done automatically or semi-automatically, therefore supporting the ground application system development and testing.

II. Role of the System Joint Test Platform (SJTP)

A. SSSGAS

The main tasks of the SSSGAS are: conduct operation control and in-orbit management based on the scientific satellite mission objectives; support earth-space joint experiments, and coordinate and schedule a network of ground stations to receive data from scientific satellites; space science data processing, distribution and archive management for scientists; achieve joint international operational management data reception and exchange; organize and carry out popular space science mission propaganda.

In general, SSSGAS is made up of Space Science Mission Operation Subsystem (SSMOS), Space SSDS (SSDS), Space Science Application Subsystem (SSAS) (multiple), and Data Receiving Subsystem (DRS).

SSMOS is the space science satellite science mission integrated operation and management center[3]. Its main task is to plan and analyze space science satellite missions according to scientific observation needs and conditions. On this basis, a viable scientific operation plan is developed and payload control commands are generated; complete satellite downstream, telecommand and control data collection and real-time processing; monitor and inspect payload in-orbit status and health status. At the same time, task-oriented analysis, observation planning, payload performance monitoring and analysis assessment generic utility services are provided to scientific subsystem and users through the network.

SSDS is the space science satellite mission data fast processing, centralized management and permanent archiving center, and also the scientific data product distribution, release, sharing, international exchange and user support center. Its main tasks are: implement scientific data fast pre-processing and visualization, generate primary and fast visual products, then distribute them timely to scientific application subsystem and users to make calibrations; science data and other auxiliary data centralized management, storage and permanent archive to ensure complete, safe and permanent usability of the scientific data; publish scientific data products and provide data application technical support to science users; carry out popular science propaganda.

SSAS is a bridge between projects and scientific users. And also the output end of the advanced scientific data products and the convening platform of scientific research results. Its main tasks are: carry out observation calls for proposals and selection and develop scientific observation plans; carry out data calibration, correction and generate advanced scientific data products efficiently; analyze and evaluate payload in-orbit performance; ground-satellite joint experiments and ground-based scientific experiments support; data distribution and user support during operation phase. Usually, a SSAS will be developed for every science satellite.

DRS's task is to receive satellite's downstream scientific data and transmit it to the scientific data subsystem according to the reception plan. Its principal activities include: data reception plan execution; downlink science data reception; baseband data collection and temporary storage; framed data generation and automatic network transmission; ground station autonomous monitoring and management, and status reporting; international interaction support.

The interface between the SSSGAS and external systems, and the interfaces within the subsystem are shown in Figure 1.

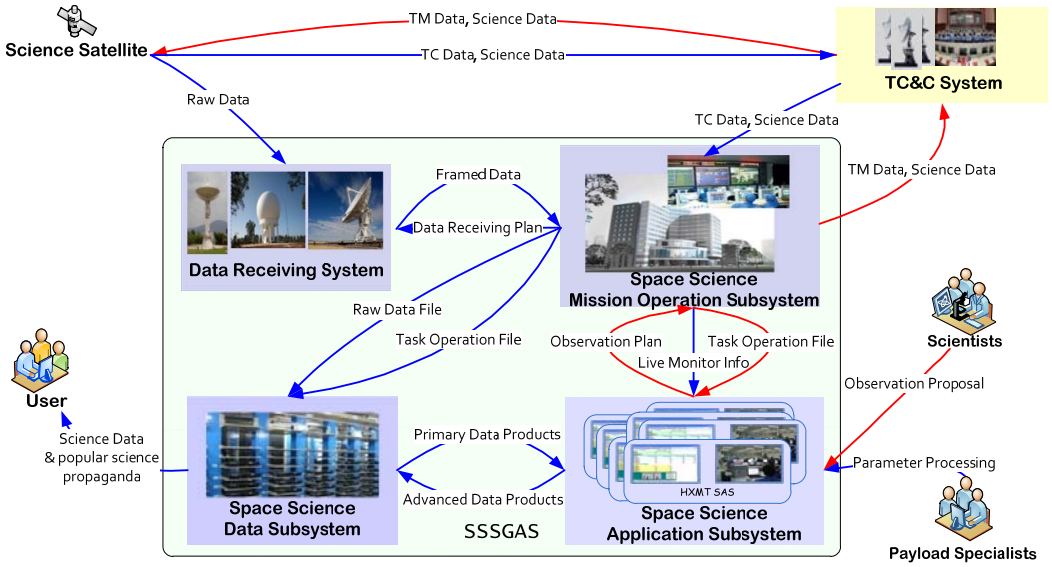

Figure 1. SSSGAS interface diagram

SSSGAS has interfaces with satellites, TC&C system, payload experts, scientists, researchers, international users and the public. These external interfaces are shown in Table 1.

Table 1 SSSGAS external interface table

No.	Interface	Source	Destination	Note
1.	Raw Data	Satellite	DRS	scientific exploration data transmitted through downstream channel
2.	Telemetry data	TC&C System	SSMOS	Satellite / payload telemetry parameters, and satellite attitude are transmitted through satellite telemetry channel; the TC&C System receive and forward the data, and also data generated by the TC&C System, such as satellite orbit and attitude data, TC&C plan reply, and instruction execution results.
3.	Observation Proposal	Scientists	SSAS	Observation proposals made by the scientists based on scientific research needs.
4.	Parameter Processing	Payload Specialists	SSAS	Payload manufacturer provides processing parameters needed by payload data processing.
5.	TC command and science data	SSMOS	TC&C system	Experimental data (not instructions) uploaded through TC&C System and varieties of instructions injected and data used to control the payloads.
6.	Science data and popular science propaganda	SSDS	Science Data Users	For scientific research team to conduct scientific research, produce scientific results, and carry out scientific propaganda to the public.

SSSGAS internal interface is the interfaces between its various sub-systems. The information of these interfaces is shown in Table 2.

Table 2 SSSGAS internal interface table

No	Interface	Source	Destination	Note
1.	Framed Data	DRS	SSMOS	Data generated after framing and adding identifiers to the data received by ground stations.

2.	Data Receiving Plan	SSMOS	DRS	Data reception plan generated by this subsystem
3.	Raw Data File		SSDS	The subsystem receives and stores the raw data file
4.	Task Operation File			Subsystem's task operation plan and its execution results.
5.	Live Monitor Info		SSAS	The subsystem sends out payload real-time monitoring information to SSASs.
6.	Observation Plan	SSAS	SSMOS	SSASs use the generic services provided by SSMOS to generate and upload science observation plans for each mission.
7.	Primary Data Products	SSDS	SSAS	The primary data products generated by SSDS are provided to SSAS for further processing.
8.	Advanced Data Products	SSDS	SSDS	SSASs generate advanced data products and send them to SSDS for archiving.

B. System Joint Testing Content

During the development and dynamic iterative process of SSSGAS, it normally takes three types of tests to complete the system joint test:

- Interface specification rationality verification test: verify the rationality and integrity of all interface definitions within the joint test environment;
- Subsystem interface verification test: after completing the subsystems, verify whether the subsystems provide the interfaces according to the specification, and whether fulfill the requirements for participating the joint system test;
- System workflow test: verify whether SSSGAS task processes are in line with the system design.

The verification and joint test have the following purposes:

- Verify system workflows and the interface specification; and optimize system workflow and interface design;
- Eliminate the differences in understanding the interface definition between the research units;
- Verify subsystem interfaces and the system workflows implemented; assist the subsystem interface testing and verification;
- Help to analyze the problems brought out by the test and try to improve system joint test efficiency;
- Minimize the impact of the manufacturing to the system in operation.

In addition, the SJTP could be used as part of the scientific satellite ground application simulation system when necessary, and participate in a simulation.

C. SJTP Requirements

To complete the above objectives, the SJTP is required to be capable of:

- simulate SSSGAS subsystems, function module interfaces and workflows;
- used to test the interfaces between the subsystems, the function modules within the subsystems;
- used to configure the form and content of the interfaces for different satellites; and has the ability to configure various components of the ground application system;
- used to test and verify subsystems' interfaces and workflows in the SSSGAS;
- used to test and verify system workflows in the SSSGAS;
- adapt to a variety of scientific satellites.

Interface testing and workflow testing are the core functionality of SJTP. The former is mainly used to verify whether the SSSGAS subsystems or the function modules have implemented the interfaces defined in the design, which includes the form and content of the interfaces; the latter composes of normal workflow branch and key exception branch tests. The normal flow branch test checks whether the normal workflow is implemented under the condition that interface receives and sends according to the requirements; the key exception branch test checks whether the workflow enters the exception handling branch as required an exception occurs.

III. SJTP Design

A. SJTP Architecture

In order to make the test platform with better scalability and easy maintenance, multitier distributed application framework is adopted, that is ”application layer / logic presentation layer / data layer” [4]. Framework details are shown in Fig. 2**Fehler! Verweisquelle konnte nicht gefunden werden.**. Modifications made in the upper layer would not have much or none effect to the underlying layers, thus minimizing the coupling within the system, and improving the system's adaptability and extensibility[5].

The application layer is at the top level of the framework, which implements the interactions between the user and the system. This layer receives user's requests. Then, it requests to extract information from the logical layer, and the information could be shown in various forms. The logical layer is the core layer that focuses on implementing various interfaces and workflow logic. It has a logic engine which processes a variety of rules. This layer accepts requests from the application layer; then with a specific interface and predefined process rules, the logic engine processes. The results are submitted to the application layer. During processing, logic layer calls data access services in the data layer. The data layer consists of data access services and data (and storage), which is responsible for system's access to data. The main role for this layer is to optimize data access and improve data access performance.

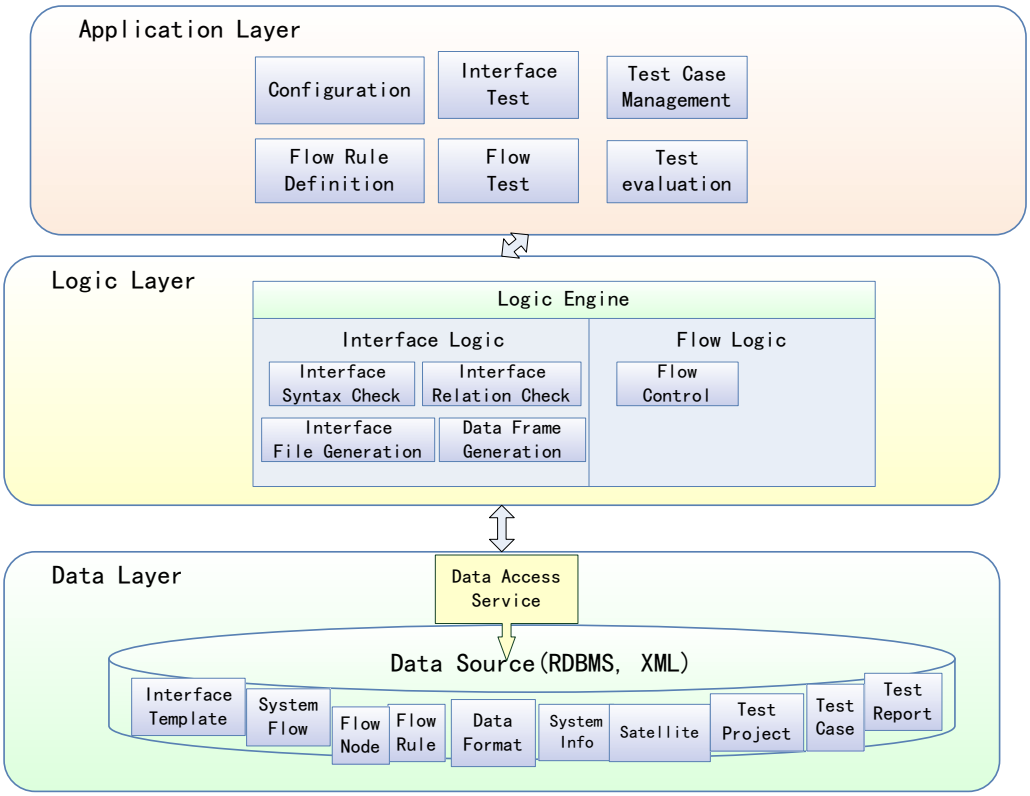

Figure 2. SJTP architecture

1. Application Layer

This layer implements human-computer interaction and external data exchange. The graphical interface for application services is provided so to better understand and locate application services efficiently. The application layer of the SJTP implemented platform related functional client application's graphic user interface and operational controls to meet user's needs on visualization.

By operating each functional client application, related processing services are called by the logic engine, and data access services provided by the data layer acquires the results, which are displayed through graphical user interface. For example, when connecting the interfaces, the application layer simulation test program is started; data access service obtains the required data, and calls the logic engine in the logic layer to achieve interface connection; the results are presented to the application layer simulation test application graphical user interface.

2. Logic Layer

Separating the system's interface and workflow logic from specific applications could provide a unified management platform for interfaces and workflows. Therefore, the system could change with specific application conditions by redefining the interfaces and workflows without making any modifications to the application layer. First, during development, we can focus on the workflow logic, thereby reducing the workload of application development and build the system quickly. Second, when the system is running and the workflow logic changes, the workflow could be redefined to adapt the changes without the need to program a new application logic, thus, contributing to business process automation and workflow reengineering.

3. Data Layer

The data layer is responsible for the definition, maintenance, access and updating data. And it also manages and fulfills application services data requests. The application layer and logical layer access data through data access services. The data layer provides unified, standardized data services through data access services. The data services are used to shield data storage, organization and access details.

B. SJTP Functional Components

SJTP achieves joint test by simulating SSSGAS's subsystems' and function modules' interface functions and workflows. Figure 3 shows the functional components of the SJTP. Using platforms and modules becomes popular nowadays[6].

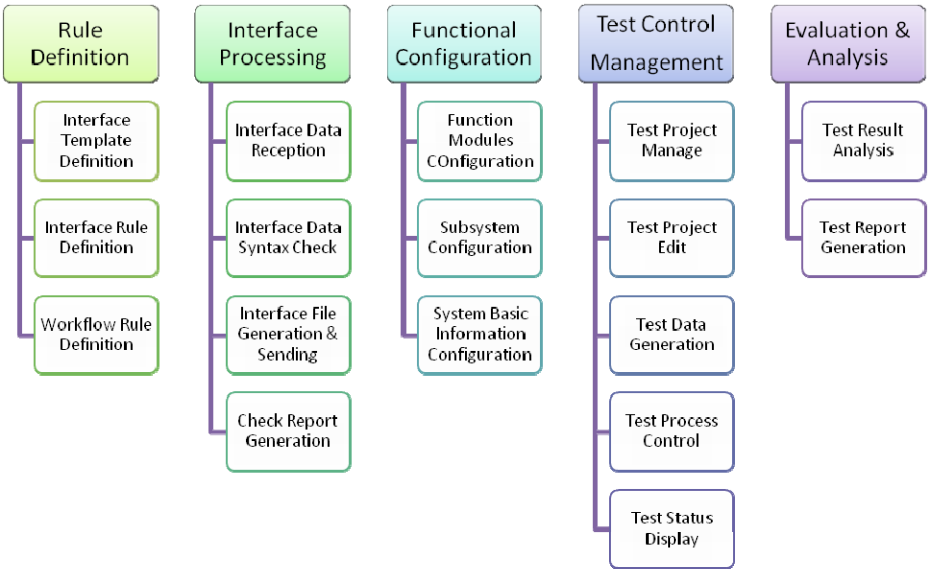

Figure 3. SJTP functional components

To simulate and test SSSGAS's interfaces and workflows, SJTP needs to have the following five functions:

- Rule Definition: Define the rules, including interface templates, interface rules, and workflow rules, required by SJTP when configuring automated SSSGAS tests.
- Interface Processing: SJTP's basic functional unit that parses and checks the interface files. And replies according to the interface specification.
- Functional Configuration: SJTP is configured as different function modules or subsystems.
- Test Process Management: Controls SJTP's testing process, including test data generation, test process control and test status display.
- Evaluation & Analysis: Analyze the test results and generate formatted test reports.

1. Rule Definition

- Interface template definition: Analyze SSSGAS according to the design to establish a common interface template. On the basis of the interface template, a variety of internal and external interfaces could be configured.
- Interface Rule Definition: According to SSSGAS's system design, map between the different types of interfaces and data, that is to acquire the reply needed after a function module or subsystem interface

processes the data received. Interface rules are a simplification of the actual function modules or subsystems.

- Workflow Rule Definition: According to SSSGAS's system design, define the SJTP test action sequences. During system workflow test, system workflows are simulated based on the workflow rules.

2. Interface Processing

- Interface data reception;
- Interface data syntax check;
- Interface file generation and sending;
- Check report generation.

3. Simulated Function Configuration

In order to accomplish interface tests with the subsystems or function modules, SJTP is capable of interfacing with them. SJTP could be configured into the subsystems or function modules needed by the joint test. The SSSGAS could manage multiple scientific satellites simultaneously and set many SSASs.

- Function module configuration: configure SJTP into several related function modules and subsystems when doing system internal testing. The content is a combination of a number of interface rules and workflow rules.
- Subsystem configuration: configure SJTP into several related subsystems when testing their interfaces.
- System basic information configuration: set the name, communication parameters, and other basic information of the functional modules or subsystems.

4. Test Process Management

- Test project management: create a new, open, save or close a test project file.
- Test project edit: define test sequence in a test project file.
- Test data generation: interface template is the basis for generating test data for certain types of interface. Interface data includes normal data and abnormal data. The normal data should cover the upper and lower boundary of the normal value, and random normal values; abnormal data includes a data exception (data item missing, data item error, data entry cross-border error, etc.), data item position error, and so on.
- Test process control: start, suspend or terminate the testing process.
- Test status display: display test process and status in the forms of text and graphics.

5. Analysis and Evaluation

- Test result analysis: analyze the test results, and give the assessment results of the tested subsystems or function modules. The results include interface correctness, flow correctness, delay processing, etc.
- Report generation: generate a report for the problems occurred in the test.

C. SJTP Operation Mode

SJTP has three operation modes: automatic, semi-automatic and manual. During the interface specification rationality verification test, the interface syntax checking and data range testing could be done automatically. The sender stores the test data in the database while sending data and the receiver stores the data interpreted into the database while processing. The interface correctness is determined by comparing the data stored at the sender and receiver. With setting the test flow beforehand, the interface flows are tested automatically.

In the subsystem interface verification test, the interface syntax checking and data value range test is mostly run under automated mode. When SJTP is the sender, the test data sent by SJTP could only be verified through subsystems' feedbacks since the subsystems being tested usually are not designed to have data-logging capabilities. The testing process will be aborted when an error occurs. Based on the error messages, the test platform checks and determines the type of error. When SJTP is the receiver, the joint test on all situations of the data sent by the subsystems could not be covered because the subsystems generally do not have the ability to send exception data and could not cover all normal data.

SJTP could set the automated and manual executions as needed. With flexible configuration, SJTP could run under semi-automatic mode so that problems are easier to find.

System workflow test is to simulate the entire SSSGAS's joint test. SSSGAS could not run automatically because it is a human-in-the-loop system. Hence, the system workflow test is done by man and machine. The system

workflow test is based on SSSGAS's application mode with manual control of the testing process. However, the subworkflows could be carried out automatically.

IV. SJTP Operation

The following interface test example shows how SJTP is operated.

Before starting the system joint tests, we need to first define the interface rules, workflow rules, message format, SSSGAS subsystems' basic information according to SSSGAS interface specification and operation workflow. All the configurations are archived using the configuration management software. The interface rules, workflow rules, and monitor information format forms the SJTP method library.

Also, test data should be ready before the system joint test starts.

A. SJTP as the Receiver

The workflow when SJTP is the receiver (see Figure 4): start to receive interface data after receiving the begin test command; check whether the interface data accords with the interface specification, show error message if not, or generate and send feedback data when it is in line with the specification; record all the data and test results, and terminate the test when test termination command is received.

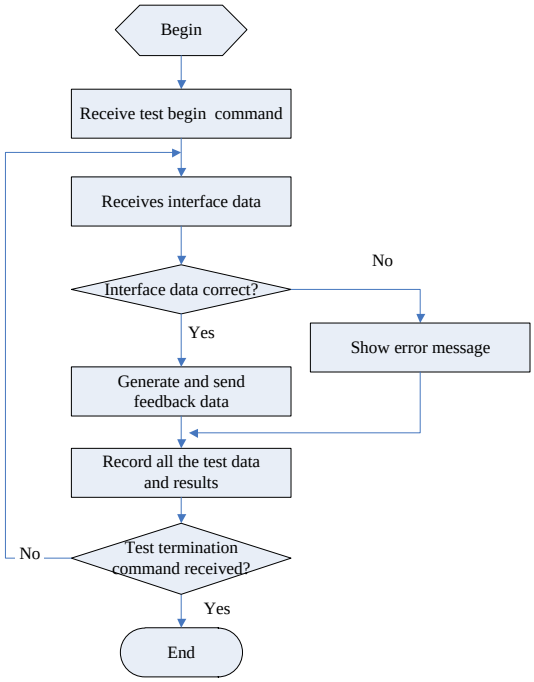

Figure 4. Workflow when SJTP is the receiver

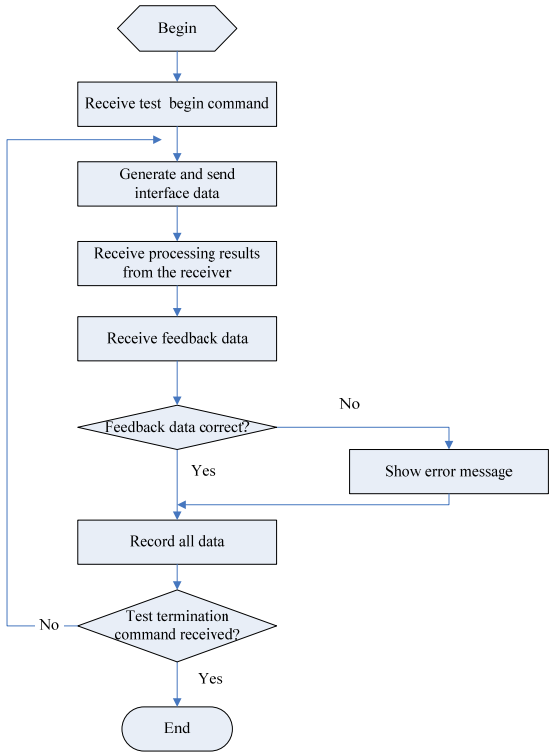

Figure 5. Workflow when SJTP is the sender

B. SJTP as the sender

The workflow when SJTP is the sender (see Figure 4. Workflow when SJTP **is the receiver** **Figure 5**): start to send (or retrieve) interface data after receiving the test begin command; record receiver processing result; receive feedback data; check whether the feedback accords with the interface specification, show error message if not, or record all data when it is in line with the specification.

V. Conclusion

SJTP proposed in this paper can be applied to the development, joint test, and verification test of the SSSGAS by simulating subsystems or function modules as needed. It greatly contributes to the automation and efficiency of the ground joint test.

References

[1]Lan Wang, “The Methodology of High Speed Railway Testing and Commissioning”, China Railway Science, Vol. 32, No. 2, 2011, pp. 104-109.

[2]Yongtao Duan, “Satellite Ground Application Testing System Design and Realization”, Master’s thesis, Beijing Jiaotong University, 2009.

[3]Rui Wang, Xiaohui Li, Jiajia Zhu and Rong Chuan, “The Business Model and Flow Design of The Ground Operational Management System of Remote Sensing Satellite”, Remote Sensing Information, No. 2, 2010, pp. 55-58.

[4]Xueqing Zhang, Wanquan Ma, Zhaohui Gao and Xiaogang Xu, “Rescarch on Resource Satellite Mission Control System Architecture”, Radio Engineering, Vol. 36, No. 5, 2006, pp. 36-38, 61.

[5]Feifei Liu, Haobo Chen, Jingyu Wang, “Rescarch on Resource Satellite Mission Control System Architecture”, Transaction of Beijing Institute of Technology, Vol. 30, No. 2, 2010, pp. 214-219.

[6]Hanying Zhang, *Spacecraft electrical testing technology*, China Astronautic Publishing House, Beijing, 2006,Chaps. 1, pp.10.