The Configuration Design for Constellation Autonomous Management Base on Multi-agent Architecture

Tao Xi*, Ruixi Gaoya, Hong Ma, Haipeng Liu and Xiaobo Hui

State Key Laboratory of Astronautical Dynamics, Xi'an 710043, Shaanxi Province, China

*Corresponding author e-mail: xxxttt2008@yeah.net

Abstract. With the construction of navigation constellation, the number of spacecrafts in-orbit is increasing rapidly, which put forward much higher requirement for the on orbit management of the satellite control center (SCC), while the conventional manual monitoring mode exists some shortcoming and is prone to meet matter. In order to overcome the disadvantage of anomaly detection, even misoperation caused by the traditional manual monitoring mode, and to improve the constellation management autonomy and intelligence level. a constellation autonomous management system software and hardware design ideas based on multi-agent architecture is presented in this paper, the whole spacecraft, sub-system and component level agent architectures is designed in term of the hierarchy ideology by employing the self-organization network configuration. A multi-agent functionality demonstration for constellation autonomous management is developed based on oriented object program design and message or event-driven mechanism, the simulation results verify the feasibility and reliability of the distribute collaborative management of the navigation constellation.

Keywords: Navigation Constellation, Constellation Management, Multi-agent Architecture, Event-driven

1. Introduction

With the rapid development of China's aerospace industry, especially the steady progress of the construction of global navigation satellite system, the number of satellites managed in orbit has increased dramatically. The ground TT&C system not only needs to complete the conventional tracking and telemetering, working condition monitoring and orbit determination tasks, but also needs to complete all kinds of control operations according to the types and needs of satellites. At the same time, it needs to continuously monitor and judge satellites, find all kinds of abnormalities in time, and carry out appropriate emergency disposal. At present, each satellite needs to invest a lot of manpower for ground monitoring work, and the manual operation monitoring mode is very prone to problems, such as the abnormal detection is not timely enough, the abnormal disposal is not fast enough, and the control misoperation is inevitable. In fact, until the mid-1980s, all space missions were managed manually by the ground control center. In order to reduce the cost of operation and management, the major organizations in the space field headed by NASA began to study the automation of some simple
and programmed manual operations\cite{[1,2]}. In recent years, with the rapid development of computer technology, artificial intelligence technology and advanced computing technology, object-oriented technology based on agent has been widely used in task planning, scheduling, modeling and control of complex systems\cite{[3,4,5,6]}. What’s more, agent has been applied to the deep space exploration-1 mission and successfully implemented the remote management, autonomous planning, scheduling and control of the detector in the United States firstly.

In order to overcome the shortcomings of manual operation, improve the emergency response capability, and reduce the operation cost of spacecraft ground system. This paper puts forward a design idea of software and hardware of constellation autonomous management system architecture based on Multi-Agent. The system, satellite and component level Agent are designed by using self-organizing network architecture, according to the idea of hierarchical design. And the function of Multi-Agent software is designed based on the object-oriented programming and message and event driven mechanism.

2. Introduction of Agent Technology
The concept of Agent first appeared in the field of distributed artificial intelligence in the 1970s. Its prototype is the Actor model proposed by Hewitt, which is a computing entity that encapsulates the address and behavior. It can communicate through message passing, and can execute actions concurrently. Actor could be anything that can perceive the environment and act on the environment through effectors. Its most basic characteristics are autonomous, responsive, pre-active and social\cite{[7,8]}. Its specific meaning is as follows:

  Autonomous: agent can operate something automatically without external intervention, and can control its own behavior and state.

  Responsive: Agent should be able to perceive the environment and respond to changes in the environment, which includes physical entity, communication network or other agent. Actually, reactivity is originally the basic attribute of organism. So emphasizing reactivity of Agents shows that Agent certain some kind of life characteristics.

  Pre-active: Agent can not only react to changes in the environment, but also be able to take corresponding actions according to its own goals. Pre-active reflects the rational characteristics of Agent in the process of decision-making and implementation behavior.

  Social: Agent can interact with other Agents or people through a certain communication language. Social can form a Multi-Agent system between Agent, thus forming the corresponding organization or society of Agent, and then realizing the abstraction, decomposition, organization and optimization of the objective world.

  Compared with object-oriented technology, Agent is a higher-granularity abstraction. In addition to the encapsulation of attributes, events and methods, it can also be endowed with thinking and decision-making capabilities, reflecting higher autonomy, greater target-orientation, flexible reactivity, and sociality of interaction with other Agents (or objects, people, etc.).

  Using Agent to solve the problem of coordinated control of complex systems can realize the distributed storage and processing of system data, control, expert knowledge and resources. Furthermore, the object-oriented Agent technology can make data resources being under a distributed control state. Data processing is usually performed locally, and only a small amount of high-level information needs to be exchanged, which is conducive to improving system efficiency.

  The general structure of Agent is shown in Figure 1. It consists of a knowledge base, an inference engine, a user interface, a communication module, an event processing module and a learning module.
The knowledge base stores knowledge of Agent; the inference engine uses the existing knowledge to control the information module, the transaction module and the learning module; the user interface can be used to protect the knowledge base and control the communication module, the transaction module and the learning module by user; the communication module is responsible for the communication with the outside world (external environment or other Agent); the transaction module contain the method for Agent to achieve the goal; the learning module summarizes the experience from the continuous operation of Agent, adds new knowledge to the knowledge base, and its learning and training can be realized through the learning algorithm of artificial scheduling network.

3. Design of Constellation Management System Architecture based on Multi-Agent

The space segment of navigation satellite consists of at least 30 navigation satellites (3 GEO satellites, 3 IGSO satellites and 24 MEO satellites), which greatly increased the amount of satellites managed in orbit. In order to improve the automation and intelligence of global constellation satellite management, at the meantime, lighten the workload and complexity of ground operation, improve the processing speed and capability, and reduce errors, a constellation management system architecture based on Multi-Agent is proposed. The system, satellite and component level Agent are designed according to the idea of hierarchical design. The system architecture is shown in Figure 2. System level Agent integrates constellation management, control and ground TT & C resources, which is the core part of Multi-Agent constellation management, considering that the constraints of TT & C conditions, there are inter satellite links between satellites.

In the navigation constellation, the system level Agent can be transplanted to GEO satellites. In addition, information interaction between 3 GEO satellites, between GEO satellites and MEO satellites, or between GEO satellites and IGSO satellites can be carried out through inter satellite link or through ground control center to realize constellation management and system reconfiguration. MEO satellites and IGSO satellites that contain satellite level Agent can realize constellation configuration management and satellite platform management. Besides, the component level Agent is mainly responsible for the management of satellite platform.
3.1. System Level Agent

System level Agent is used to improve the automation level of satellite management, allocate the TT & C resources reasonably, realize the orderly generation of constellation system management plan, automatic TT & C and application of ground TT & C resources and inter channel resources, and achieve the goal of no need to complete TT & C tasks under normal circumstances. In order to realize the autonomous management of navigation constellation, the system level Agent is embedded in the GEO satellite flight control software to form the constellation self-control layer (as shown in Figure 3), carry out constellation mission planning, collection and processing of constellation internal data, constellation state monitoring and system reconstruction.

(1) The main control unit is responsible for making the overall plan and coordinating the work of each Agent, that is, finishing the satellite planning and scheduling. Among all Agents, it has the highest degree of intelligence. The main control unit designed in this paper is shown in Figure 4. The upper layer is abstract planning and scheduling, and its main task is to determine the execution time of the flight mission. After the mission is determined, the main control unit sorts the flight missions and determines the execution time of each mission according to the situation of satellite orbit ground coverage, comprehensively considering the importance of the mission, the deadline of the mission, the completion time, and sunshine conditions, etc. Due to the limited capacity of onboard memory, it is necessary to reserve a certain execution time for data transmission and some conventional tasks in the sorting process. The sorting task can be realized by traditional planning method or expert system, and neural network with learning function.

Figure 2. The constellation management system architecture based on Multi-Agent

Figure 3. The constellation self-control layer of the system

Figure 4. The main control unit
(2) The intelligent decomposition, coordination and execution module is the middle layer between the main control unit and the real-time control module. After the main control unit formulates the flight plan, each Agent receives its own part of the overall plan, which cannot be directly executed and must be further processed. The intelligent decomposition, coordination and execution module decomposes these plans into executable control instructions. The precondition of synchronous processing and coordinating the execution time of each task is to complete the further management of resources and equipment in the system.

(3) The real-time control software and hardware are the bottom control part of the satellite, which can be divided into two kinds: one for responding to the control command, completing the corresponding functions after receiving the control command; and the other for responding to the changes of the external environment and handling the abnormal situation.

(4) The state monitoring, fault diagnosis and system reconfiguration module monitors the control commands received by the real-time control software and hardware, infers the state of the system when executing these commands by using the satellite model and algorithm, and then compares with the information collected by the sensor. If the information collected by the sensor is consistent with the normally executed commands in the inference model, it will be considered that the command has been executed correctly; otherwise, it will be considered that the system is abnormal. When a fault occurs, it is necessary to identify the specific faulty component and the mode that causes the fault according to the sensor information and the reasoning model; if the fault cannot be eliminated, it will reset the system. At last, the execution state of satellite command is transmitted to the upper module, and the system reconfiguration state is transmitted to the main control unit to update the status, which is used as the basis for further tasks.

The above three modules constitute the Agent system that satellite autonomously controlled. At the same time, it retains the control ability of the ground control system. The ground control system can intervene in the autonomous control through multiple levels, such as modifying the flight mission, modifying the flight plan generated by the planning and scheduling module, controlling the real-time control software and hardware of the satellite directly, etc. What’s more, the ground controller can issue instructions to control the underlying software and hardware of the satellite directly, and obtain the telemetry information through the condition monitoring module. It constitutes the traditional satellite TT&C system, and can completely replace the autonomous control when needed.

3.2. Satellite level Agent

Satellite is a complex system, which is composed of different subsystems such as TT&C, attitude and orbit control system, payload, satellite management, thermal control, energy, propulsion system and inter satellite link. These subsystems have relatively independent functions and can complete their own sub tasks. However, these tasks are quite different. For example, the attitude control subsystem mainly completes two functions: attitude measurement and attitude adjustment, and the temperature control system is mainly responsible for the temperature monitoring and adjustment of onboard sensors. In addition, the satellite is an integrated body, and each subsystem is interconnected. They must cooperate with each other to effectively complete the task. For instance, when the payload works, it will make a request to attitude pointing, and needs to occupy certain on-board resources, which requires the cooperation of each subsystem. The satellite level Agent is shown in Figure 4.
3.3. Component Level Agent

The component level Agent covers the components of all subsystems of the satellite:

1. Attitude control Agent is used for satellite attitude measurement, attitude acquisition and attitude control;

2. The orbit control Agent measures the position of the satellite, calculate and maintenance orbit using GPS (if there is), or through injecting orbit on the ground;

3. The thermal control Agent measures the temperature of each monitoring point of the satellite, and adjusts the temperature according to the demand;

4. The energy system Agent manages the energy parts of the satellite, such as solar array and battery;

5. The payload control Agent monitor and control the payload. When there are multiple payload, multiple Agents can be set up;

6. Data transmission Agent is responsible for data management and transmission;

7. Inter satellite link agent gives the measurement status of inter satellite communication, including inter satellite link establishment status, inter satellite ranging information, etc

8. The TT&C system Agent can receive the ground control command, downstream the telemetry, ranging and angle measurement information of satellite.

3.4. System Integration and Implementation

In the long-term operation of global navigation constellation, a single satellite in the constellation will deviate from the nominal working orbit due to the influence of the earth's non spherical gravity, light pressure and other stars’ gravity. In order to ensure the accuracy and reliability of ground applications, the ground control center needs to implement the corresponding control management. In this section, the system working process is given to verify the effectiveness of constellation autonomous management framework based on Multi-Agent that proposed in this paper, taking the longitude maintenance of MEO satellite’s regression cycle as an example. The simple working framework is shown in Figure 6.
Figure 6. Constellation management architecture of GEO_Agent

Assuming that the main control Agent unit of each satellite in the constellation can obtain the measured value or condition monitoring value of the component level Agent in real time; the Agent of three GEO satellites in the navigation constellation (GEO_Agent) have the global control function of the constellation; there is an inter satellite link between the MEO agent (MEO_Agent), IGSO agent (IGSO_Agent) and the GEO agent (GEO_Agent); the ground TT&C center can intervene the Agent to communicate with the GEO_Agent all time, which ensures that the longitude maintenance of MEO satellite’s regression cycle result can fed back to the ground through the GEO_Agent.

In a certain TT&C period, if the orbit control Agent measurement of the No.2 MEO satellite on the second plane (ID_P2_MEO_2) of the global navigation constellation shows that the longitude deviation of the return period exceeds the designed longitude, the message generated by the orbit control Agent (Msg_DO_ID_P1_MEO_2) will be sent to the corresponding GEO_Agent_i (i=1,2,3) through the MEO_Agent and the inter satellite link Agent to realize the message distribution (loop message). Then, Geo_Agent_i will determine whether to respond to the message according to the priority of the message queue (MessageQueue). If there is no message in the message queue or the priority of other messages is low, GEO_Agent_i sends the message to the MEO_Agent to response. Otherwise, it waits. If the GEO_Agent_i sends a message to the ID_P2_MEO_2 satellite to response, instantiates the message handler thread (HandlerThread) and prepares to process the message, the main control Agent unit of the MEO Satellite distributes the message to the orbit control Agent, and sets the state before orbit control by the sub thread Skill_DO; the Attitude Control Agent is dispatched to adjust the ignition attitude by the sub thread Skill_AOC; the propulsion Agent is dispatched to prepare for ignition by the sub thread Skill_Prop. When the longitude control condition of regression period is satisfied, the MEO_Agent sends the ready message (Msg_OK). The Geo_Agent forwards the message sent by the MEO_Agent(Msg_OK_ID_P1_MEO_2) to the ground control center through the ground intervention Agent for status confirmation, and then sends the ignition control message if it is correct. After the ignition control, the MEO_Agent dispatches the orbit control Agent to measure the controlled orbit in real time. If the control requirements are fulfilled, the autonomous control process of the constellation ends, otherwise the above process is repeated. The processing, distribution, and queue instantiation of message are shown in Figure 7.
4. Conclusion

To fulfill requirements of long-term management automation of navigation constellation, the constellation autonomous management system based on Multi-Agent framework is studied. Then, the system level, satellite level and component level Agents are designed according to the idea of hierarchical design. Furthermore, the current popular Android embedded operating system is integrated into navigation satellites, which adopts event driven and message processing mechanism. In addition, the message processing and distribution mechanism between navigation satellites is briefly given based on object-oriented Java language. Last but not least, the feasibility of constellation autonomous management system framework proposed in this paper is given, taking constellation control as an example, which makes a certain technical reserve for the subsequent design of constellation long-term management automation system.

References

[1] Jim Taylor, Michela Deep Space 1 Telecommunications NASA, Jet Propulsion Laboratory, 2001
[2] Sarka D M et al. The real-time Object Agent software architecture for distributed satellites systems. In: Proceeding of 2001 IEEE Aerospace Conference (6). Piscataway: NJ IEEE Inc., 2001.2731-2741
[3] Schetter T, Campbell M, Surka D. Multiple agent-based autonomy for satellite constellations. Art. Int., 2003,145:147-180
[4] Macal Charles M, North Michael J. Tutorial on Agent-based modeling and simulation[C] //Proceeding of the 2005 WSC Orlando, FL, USA: the 2005 WSC, 2005:2-15.
[5] D.M. Surka, M.C. Brito, and C.G.Harvey. The Real-Time ObjectAgent Flight Software
Architecture for Distributed Satellite Systems, 2001 IEEE Aerospace Conference Proceedings, Big Sky, Montana March 2001.

[6] Laszlo Gulyás, Tamás Kozsik, The Multi-Agent Modelling Language and the Model Design Interface, IDEAS/RePEc, 1999

[7] G. Weiss), Multiagent System: A Modern Approach to Distributed Artificial Intelligence, Cambridge, MA: MIT Press, 1999.

[8] J. M. Bradshaw, Software Agents, Cambridge, MA: AAAI Press/MIT Press, 1997.