

# A C++ Based Modular Simulation Software of Satellites Earth Observation Mission Planning

Xiaoheng Liang<sup>1</sup>, Yang Gao<sup>2,a</sup> and Shenggang Liu<sup>2</sup>

<sup>1</sup> Beijing Institute of spacecraft system engineering, Beijing, China

<sup>2</sup> School of Astronautics, Beihang University, Beijing, China

**Abstract.** A simulation software for efficient analysis and rapid planning of multi-satellite earth observation mission is designed to solve the problem that it is difficult to make fast decisions for multi-target and multi-constraint earth observation mission. The software is developed with C++ programming language, which has the advantages of modular design, good portability, fast running speed and less occupation of resources. Meanwhile, the input and output interfaces of the software adopt the file format of an extensible mark-up language, which can easily realize cross-platform (Windows, Linux, etc.) data exchange. The simulation results show that the software has high execution speed and precision.

## 1 Introduction

Earth observation plays an important role in national economic and social development (such as land census, emergency and disaster relief, etc.) and is an important part of a country's comprehensive strength<sup>1,2</sup>. However, satellite earth observation and mission planning design are a very complex engineering activity<sup>3</sup>.

Satellite mission planning is based on satellite resources to generate scheduling sequences oriented to observation tasks and satisfying various constraints. Although the number of satellites in orbit is still increasing, the limited resources of earth observation satellites are extremely precious compared with the growing observation demand. It is a more urgent problem on how to make overall planning of satellite resources and accomplish corresponding task demands with minimum resources. Mission planning of earth observation satellites plays an increasingly important role in satellite control. At the same time, the cost of most satellites has already exceeded 100 million dollars, because satellites contain a large number of complex and sophisticated technologies that are difficult to build. To make better use of satellite resources and bring them into full play in the military and national economy, it is necessary to use appropriate mathematical models and software tools to manage and distribute satellite resources and develop optimized satellite imaging plans<sup>4</sup>.

Besides, there are many constraints on earth observation missions<sup>5</sup>, such as sunlight, resolution, power, weather constraints, and payload type constraints. As the number of satellites, facilities and observation objects increase, the complexity of mission planning increases geometrically. Therefore, faced with a large

amount of satellite, facilities and various constraints of observation targets, it is difficult for commanders to make decisions quickly according to the requirements.

Given the above problems, a number of institutions have researched the mission planning of earth observation satellites. However, most of these studies only involve single-satellite strategy (or roughly decomposing multi-satellite tasks into single-satellite tasks) and observation tasks with relatively simple constraint conditions<sup>6</sup>, and there are few observation constraints to be considered in the task modeling process. As a result, the simulation of the real environment of the satellite and the target cannot be carried out well. Therefore, it is necessary to develop the mission planning simulation software with strong modeling ability and efficient running speed.

Multi-satellite earth observation mission planning and simulation are based on the spacecraft dynamics theory, including attitude maneuver constraints, data storage constraints, power control constraints and dozens of other constraints, to achieve the multi-satellite earth observation mission efficient analysis and rapid planning of simulation software. The simulation platform can not only be applied to the simulation of satellite earth observation mission planning and its attitude maneuver strategy but also can provide certain support for the analysis, design, and evaluation of large systems, enabling users to make decisions quickly based on the planning results. Also, for different types of observation targets (targets, area targets and trajectory targets), the planning strategy adopted by the mission planning simulation software will be optimized accordingly.

Firstly, this paper introduces the main process of mission planning simulation software. Then, the

<sup>a</sup> Corresponding author: 47178589@qq.com

composition and important functions of the software are described in detail. Finally, combined with the case test of multi-satellite to target and area target, the simulation ability of the software to the actual task is demonstrated.

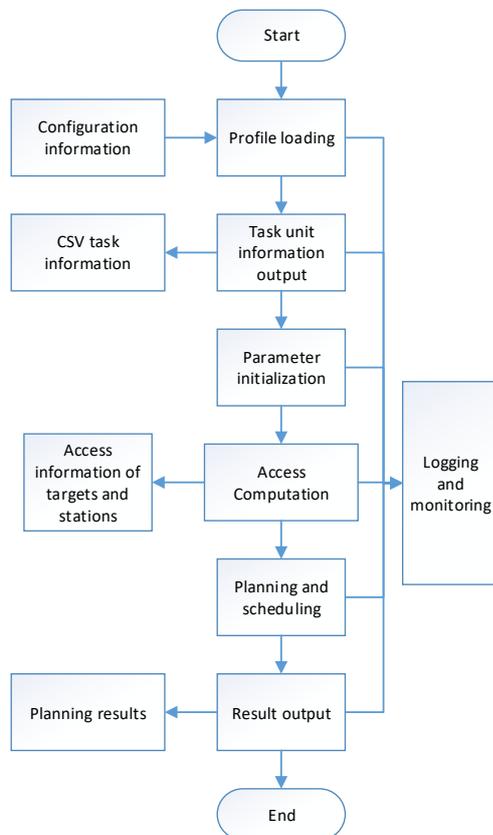
## 2 Mission planning process

Mission planning simulation software mainly consists of configuration file loading module, unit information output module, parameter initialization module, unit access computation module, planning and scheduling module, planning result output module and task log module.

The program is developed with C++ programming language, which can easily maintain and expand. At the same time, commonly used data files are replaceable.

The input and output file format of the software is the Extensible Mark-up Language (XML). The simplicity of XML makes it easy to read and write data in any application, which quickly makes XML the only common language for data exchange <sup>7</sup>. Since the program is developed in C++, using XML as an input and output format means that the program can easily exchange data with Windows, Linux, and other platforms, as well as make the software portable.

The running process of the mission planning simulation software is shown in Figure 1.



**Figure 1.** The execution process of mission planning

Firstly, the program loads the input XML file configured by the user in the interface format defined by the mission planning software. After the configuration file loading module completes the parsing of the input

XML, the unit information output module generates a Comma Separated Values (CSV) file for user reference. Then, the parameter initialization module will establish the information matching between the satellites and the observation targets (including the facilities). The unit access computation module divides the output information of the previous module into meta-tasks and calculates the visibility of satellites to the targets and facilities. Finally, after the meta-tasks are planned and combined according to the strategy by the planning and scheduling module, the planning result output module writes the observation plans into the output XML file to finish the process.

The task log module can monitor the key steps and record the problems encountered during the whole process of mission planning.

## 3 Software construction and functions

### 3.1 Input and output

Before starting the task planner, the user needs to input requirements for the simulation task into an XML file in the mission planning interface definition format. This file contains information about the step size, total simulation duration, satellite, observation target and facilities of the mission simulation.

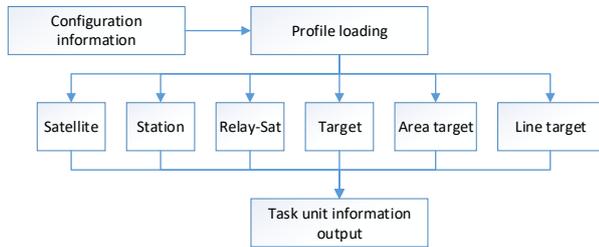
The partial nodes of the input XML are shown in table 1.

**Table 1.** Nodes of mission planning input XML

Nodes	Name	Fields
1	task attributes	clock
1.1	task interval	interval
2	satellites' info	satellites
2.1	orbit-propagator	propagator
2.2	sensors' info	sensors
2.3	satellites constrain	constrains
3	stations' attributes	stations
3.1	minimum of elevation	minElevation
4	targets' info	targets
4.1	sensors' type	sensorType
5	area targets' info	areaTargets
5.1	observation frequency	imageTimes
6	trajectory targets' info	lineTargets
6.1	observation time	imagePeriod

After the mission planning starts, the configuration file loading module first extracts the information of the input XML file by using the parser. Then, based on the extracted information, the corresponding instance is generated and stored in the memory unit to prepare for

subsequent module calls. The relevant information involved in the configuration file includes mission information, satellite information, relay satellite information, earth station information and various observation target information. The execution flow of this module is shown in Figure 2.



**Figure 2.** The process of loading configuration file

After completing the planning and scheduling of the observation task, the output module of the planning result will generate an XML file according to the output interface format of the mission planning simulation software. This file contains the task attributes, target types, load attributes, start and end time of the task, attitude maneuver angle and working time, instruction loading information and data download information of the observation task sequence. Some of the nodes that output XML are shown in table 2.

**Table 2.** Nodes of mission planning output XML

Nodes	Name	Fields
1	task attributes	clock
1.1	plan results	ifSuccess
2	satellites' info	satellites
2.1	plans' info	obsvPlans
2.1.1	targets' info	target
2.1.2	sensors' info	sensor
2.1.3	attitude maneuver info	attiMnv
2.1.4	commands upload info	cmdUpload
2.1.5	data download info	dataDownload

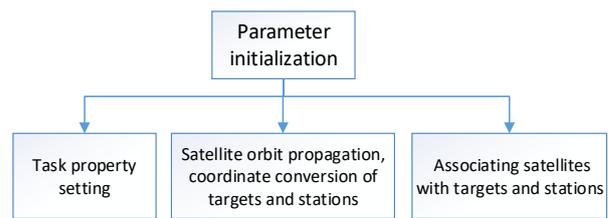
### 3.2 Task unit information output

The XML file has the advantage of data exchange and specification unification, but it cannot show the information of the task unit visually. The unit information output module uses input XML file information to output CSV files corresponding to task unit information according to different categories of task units (such as satellite and facilities). CSV is a common, relatively simple file format that is most widely used to transfer tabular data between programs<sup>8</sup>. The CSV file can display information about the same type of task unit, and the user can view the CSV file to see if the input is the same as the actual requirement.

### 3.3 Task initialization and modeling

After loading the input information, the whole earth observation mission needs to be modeled. First, the parameter initialization module sets the initial parameters of the task and establishes relevant constraints. For example, the total simulation time of the task, simulation step size, start and end time of the task scene, etc. Then, the module needs to forecast the orbit of the satellite and convert the coordinates of the earth station and observation target. Finally, the parameter initialization module establishes the communication matching between the satellite and the observation target, the earth station and the relay satellites.

The main flow of the parameter initialization module is shown in Figure 3.



**Figure 3.** The initialization process of parameters

Before the simulation, some limitations/constraints need to be completed, such as the limit of the range of the satellite's attitude maneuvering ability, the minimum visible elevation of the satellite to the observed target, and the maximum number of satellites that the earth station can connect to at the same time. Establishing these constraints can improve the simulation of the real task environment.

Satellite orbit modeling is an important process of mission planning. According to the satellite information in the input XML, the orbital forecaster makes ephemeris forecast for the satellites in the mission scenario. Mission planning simulation software currently supports two-body, J2, SGP4, and High Precision Orbit Propagator (HPOP).

The description of the position parameters of targets, area targets, and general trajectory targets on the earth's surface is usually based on the earth-centered earth-fixed (ECEF) frame, but the coordinates in the earth-centered inertial (ECI) frame need to be used in subsequent access computations. This module has three coordinate conversion functions. Among them, the accuracy of the fast coordinate transformation function is similar to the conversion function provided by IERS, and the speed is much faster than the latter, which can significantly shorten the time of coordinate transformation and the whole task simulation process.

The location of the earth station is generally described by geographical coordinates, and its coordinates in the inertial reference frame are needed in the access computation. The difference between the coordinate transformation of the earth station and the observation target is that the geographical coordinates need to be converted to the geocentric reference frame coordinates

first, and then to the inertial reference frame coordinates later.

In a multi-satellite to earth observation mission, each satellite may have different observation targets. In the parameter initialization module, the observation target configuration function matches each satellite and its target to be observed. In the subsequent access computation module, the program calculates only the position relationship between the satellite and the matching target. If the user limits the facilities and relay satellites that the satellite can use, the corresponding constraint information can be written in the configuration file. The module will then establish a constraint relationship between the satellite and the measurement and data transmission resources that can be invoked. If relevant constraint information is not written, the mission planning simulation software, namely the facilities and relay satellite with default input profile, can communicate with the imaging satellite.

### 3.4 Access computation

In earth observation, the satellite can only start to perform the observation mission when the observation target appears in the field of view of the payload (such as CCD, infrared imaging and SAR, etc.).

The unit access computation module uses the satellite forecast ephemeris generated by the parameter initialization module and the coordinates of the observation object and the measurement and control facility (earth station and relay satellites) in the inertial reference frame as input and then calls the module function to calculate the geometric relationship between the two.

The results are the visible windows (visible start and end times) between the mission elements (satellite and observation target, satellite and facility). The access computation is used for planning and scheduling module calculations, and it is output to a text file.

Before the geometric calculation, the access computation module will divide each task into units based on the time sequence, and the length of the units is based on the simulation step size of the whole task (which can be specified by the user). Targets access windows divided by time sequence are called "meta-tasks". The meta-task computing method can reduce the complexity of visibility computing and improve the efficiency of the whole mission planning process.

The access computations for satellite point-to-targets and facilities are similar, so these calculations can use the same functions. The results of the satellite to point target access computation include the start and end time of the visible arc, the attitude maneuver angle, the load type used in the observation and the resolution. The results of satellite access computations for facilities include the start and end times of the visible windows, the available duration, and the facilities (or relay satellites) number.

The visibility of the satellite to area target and general trajectory target is different from the above calculation. Before calculating the visibility of the region target and

the general trajectory target, the two types of targets need to mesh, and the segmentation accuracy is specified by the configuration file entered by the user. The results of the satellite's access computation of area targets include the start and end time of the visible arc, the visible boundary, the load type used in the observation, the coverage rate and the resolution, etc.

For satellites with attitude maneuvering capability, it is necessary to increase the calculation of attitude side swing angle during the calculation of visibility (and observation target). If the satellite needs to adjust the orientation of the imaging load utilizing lateral maneuver to complete the imaging of the observation target, it is necessary to calculate whether the attitude maneuver time is within the allowed time window.

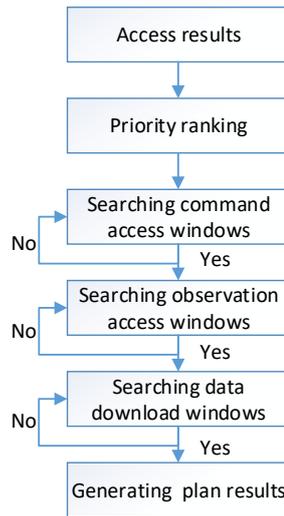
### 3.5 Planning and scheduling

There are three main conditions for the earth observation mission to be carried out. Firstly, the satellites should receive the loading instructions from the facilities. After that, the satellites are able to observe the specified target. Finally, data can be exchanged between satellites and designated data transmission facilities. Satellites that meet the above three conditions in time order can achieve a complete planning response.

The observation task arrangement is a complex process with repeated iterations. The program needs to calculate whether the first command loading window of each satellite and the designated observation and control station (if not specified, it needs to calculate all stations) conflicts with the first window visible to the observation target (if there is a conflict, it needs to find the closest non-conflict window). Then, it is necessary to select the nearest digital transmission window in the visible window of the observation target satisfying this condition. The data transfer window also needs to be wide enough for the satellite to upload all the generated data. If there is a conflict between before and after task scheduling, the priority of the meta-task sequence should be arranged according to the corresponding strategy. If a suitable one is not found in all the windows, the plan will be deemed ingenerated.

The multi-satellite planning includes many constrains such as optimizing the selection of the imaging orbit of the observation strip, evaluating imaging quality corresponding to the imaging time of each orbit and the time planning of the carried out observation strip. In the mission planning, it is possible to roughly estimate the duration of observation, data transmissions, the conversion time of satellite attitude and the changes of relevant on-board state parameters.

There are three different planning strategies for multi-satellite earth observation mission planning and simulation: the shortest mission response time, the best imaging quality and the highest mission completion. The main flow of the planning and scheduling module is shown in Figure 4.



**Figure 4.** The process of planning and scheduling

The shortest response time strategy is the time-optimal strategy based on the first observation results of the satellite. Task requirements have different priorities for different observation targets. Therefore, the mission planning simulation software puts forward the concept of "target observation weight". The shortest response time is to make the task response time of each target as early as possible under the premise of the priority target observation weight.

The optimal imaging quality strategy is based on the resolution of the satellite observation target. The optimal imaging quality refers to giving priority to the target with "the highest imaging quality" and making the imaging resolution of each target as high as possible under the premise of the priority target's observation weight.

The strategy with the highest mission completion is the strategy with the highest number of imaging times based on the satellite observation mission frequency. The task completion rate is the highest, which means that under the premise of the observation weight of the priority target, the number of imaging times of the observed target should be as much as possible, and the task with the highest observation frequency should be first met (single-target task completion rate = the number of observation schedules of the target/the number of observation times of the target).

### 3.6 Logging and monitoring

Logging is one of the most important aspects of software development. It can help quickly locate problems. The task log module is used to record the exceptions during the whole mission planning simulation cycle. When the mission planning simulation software runs unexpectedly, the module will call the task log module to write the location (or possible cause) of the error to the log file.

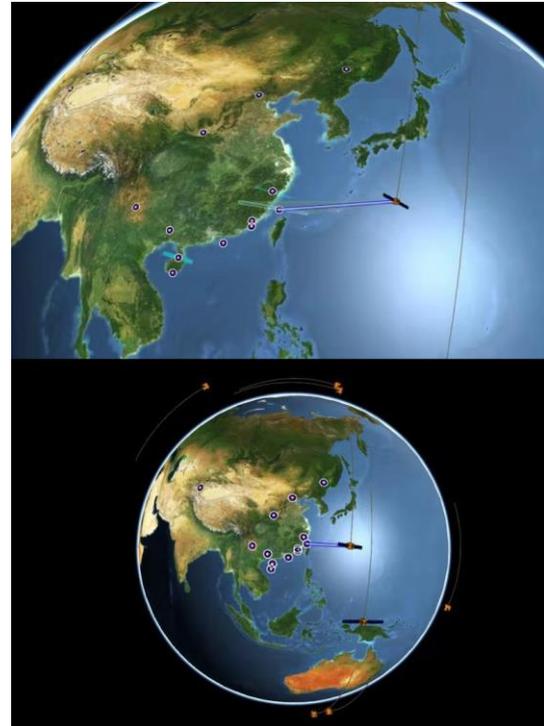
## 4 Application examples

In earth observation missions, the typical objects to be observed are targets and area targets. To demonstrate the

simulation capability of the mission planning software, this paper designs the targets and area targets cases respectively for test.

### 4.1 The targets example

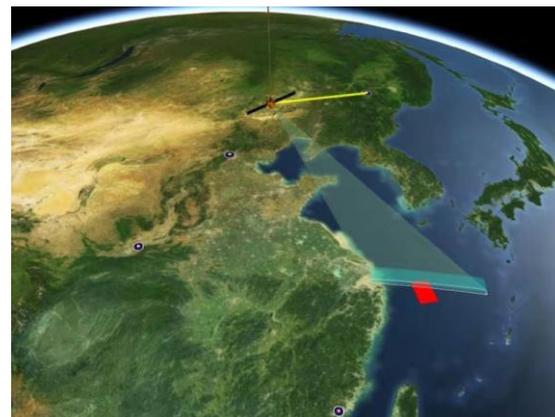
7 satellites and 13 targets are used in this example. The mission time of the simulation is 1 day. The process of the observation in the simulation is shown in Figure 5. The whole running time that this simulation example takes is less than 0.75 second.



**Figure 5.** Multi-satellite observation for targets

### 4.2 The area target example

In the case of the area target, 10 satellites, an area target and 8 facilities are used as task units in this paper. The mission time of the simulation is 1 day. The process of the area target observation is shown in Figure 6. The whole running time that this simulation example takes is less than 7.03 second.



**Figure 6.** Multi-satellite observation for area target

## 5 Conclusion

To solve the problem that multi-satellites, multi-object, and multi-constraint earth observation mission planning is difficult to make a quick decision, this paper designs a modular simulation software based on C++ language.

The simulation software can apply dozens of constraints such as solar elevation angles, power supply, weather information, data transmission and storage to satellites, facilities and observation targets to simulate the real environment of many earth observation missions. The modular design makes the software easy to debug and clear in structure. The task log module can provide quick feedback for software running and troubleshooting. Adopting XML file as input and output interface format has the characteristics of standardization, user-friendliness and cross-platform, which can provide good support for data exchange. Software development makes full use of the advantages of C++ language to achieve maximum computing efficiency and resource utilization. Besides, the author continues to explore parallel computing methods, which can further improve computing speed.

## References

1. Jiang Wei, Hao Hui-cheng, Li Yi-jun. Review of task scheduling research for the Earth observing satellites[J]. *Systems Engineering and Electronics*, 2013, **35**(9): 1878-1885.
2. Gao Hong-tao, Chen Hu, Liu Hui, et al. Development of Overseas Earth-Observing Satellite Technology[J]. *Spacecraft Engineering*, 2009, **18**(3):84-92.
3. Bonnet J, Gleizes M P, Kaddoum E, et al. Multi-satellite Mission Planning Using a Self-Adaptive Multi-agent System[C]// *IEEE International Conference on Self-adaptive & Self-organizing Systems*. 2015.
4. He Ren-jie, Gao Peng, Bai Bao-cun, et al. Models, algorithms and applications to the mission planning system of imaging satellites[J]. *Systems Engineering —Theory & Practice*, 2011,**31**(3):411-422.
5. Tavana M , Hatami-Marbini A . A group AHP-TOPSIS framework for human spaceflight mission planning at NASA[J]. *Expert Systems with Applications*, 2011, **38**(11):0-0.
6. Sun Kai, Xing Li-ning, Chen Ying-wu. Agile earth observing satellites mission scheduling based on decomposition optimization algorithm[J]. *Computer Integrated Manufacturing Systems*, 2013, **19**(1):127-136.
7. Calvanese D, Giacomo G D, Lenzerini M. Representing and Reasoning on XML Documents: A Description Logic Approach.[J]. *Journal of Logic & Computation*, 2018, **9**(3):295-318.
8. Wang N, Parthasarathy S, Tan K L, et al. CSV:visualizing and mining cohesive subgraphs[C]// *Acm Sigmod International Conference on Management of Data*. 2008.