Software Tools for Modeling Space Systems’ Equipment
Command-and-Software Control

Ludmila F. NOZHENKOVA, Olga S. ISAEVA and Alexander A. EVSYUKOV

Institute of Computational Modelling of the Siberian Branch of the Russian Academy of Sciences,
Akademgorodok 50/44, 660036 Krasnoyarsk, Russia

Keywords: Simulation model portability, Spacecraft, onboard equipment, Command and measuring system, Simulation modeling.

Abstract. The article describes the instruments allowing to create software models of the space systems’ onboard equipment function. We have developed the original methods and tools allowing to build information-and-graphical models describing the onboard systems’ architecture. The model function are set in the knowledge bases and allow to describe different variants of the onboard equipment command control. We are working within the area of creation of software for the problem-oriented simulation infrastructure based on the Simulation Model Portability standard. Our software complex will allow an onboard systems’ designer to conduct simulation tests for preparation and analysis of technical projects.

Introduction

The technologies of system analysis and computer modeling are an essential part of the modern technical science. Testing of complex technical systems, as a rule, encounter economic and technological difficulties. That is why we observe increase of the importance of the software tools allowing to simulate the systems until the stage of their manufacturing and use the built models for equipment design and operation later on. In order to simulate the objects with their own functional logics, consisting of the subsystems of different purpose and manufactured by different companies, it is necessary to focus software tools on the work in a uniform international standard – the Simulation Model Portability (SMP2) [1]. The SMP2 standard determines the universal approaches to organization of the simulation systems, provide usage of the simulation models and their transferability in the simulation environments. It doesn’t regulate the ways of model building, but sets the rules of interaction of different models and is a platform for their integration and joint use in big complex projects.

The SMP2 standard is widely used in the tasks of the space systems’ modeling. For example, it is the base for the simulation infrastructure SimSAT, designed by the European Space Agency [2], the mission control center simulator on the basis of SWARMSIM [3], South Korean Aerospace Design Institute’s software [4], etc. Although the approaches are common, each of the above examples uses original methods for model building.

For creation of the original simulation infrastructure [5] we have suggested software tools allowing to build simulation models of the space systems’ onboard equipment function. We have presented the methods of information-graphical and intellectual modeling [6], and the method of model integration [7]. The information-graphical modeling method allows to set the system’s structure in a graphical form, visualize transmission of commands and impulses during simulation, make telemetry generation and analysis. The intellectual modeling method allows in an interactive mode to create the rules of the model work describing the logics of the onboard systems’ interaction in the terms of the subject area. The integration method allows shared usage of the models in SMP standard with intellectual rule-based models and virtual tools. Application of the software tools has been studied for the purpose of the onboard equipment command-software control modeling. [8].

87
Information-graphical Modeling Tools

The purpose of modeling of the onboard equipment’s command-and-software control is to simulate function of spacecraft systems during command receipt and execution. The simulation algorithm can be described as a sequence of software imitator’ actions. Ground control complex software simulator transmits a command to spacecraft onboard equipment command-and-measuring system’s simulator. The command-and-measuring system analyses the command and generates a receipt in telemetry. On the basis of the analysis, the command is either executed or transmitted to the onboard control complex software simulator. The onboard control complex executes the command, creates a respond and adds it to telemetry. Command execution changes the state of the equipment and it is also shown in telemetry. The onboard remote signaling equipment is responsible for formation of telemetry, command-measuring system and reception-transmission devices are responsible for its transmission. On the basis of the analysis of telemetry an output is formed about the command transmission and execution, and also about the state of the onboard systems. In order to provide unification of data, the European Space Agency standards are used: ESAPSS-04-107 describes telecommand format, ESAPSS-04-106 – telemetry format. Later on, the results of modeling can be used during spacecraft onboard equipment ground testing.

For this model creation, we suggest to use information-graphical modeling tools. Such approach can be found in the existing simulation infrastructures. They use such graphical and non-graphical modeling tools that create C/C++ code. For instance, the modeling language Modelica [9] allows to create structural aspects of a model and then determine model’s behavior with the help of the graphical modeling tool such as Dymola. The compiler Modelica generates C++ code of a model. Popular modeling systems such as AnyLogic, MATHLAB Simulink, also include graphical tools. However, their models, as a rule, do not represent the real structure of a modeled system due to the necessity to use additional functional blocks designed for supporting of the processes of modeling, visualization, data entry and output. Such tools are of a universal character which makes it difficult to use them in a problem-oriented environment.

We suggest the graphical modeling tools that are most complete for the specifics of the tasks under solution. Model building consists of a visual presentation, configuration of model’s blocks being a hardware part of the onboard systems, and of setting commutation interfaces, relations and parameters of the modeled equipment. Figure 1 is an example of graphical presentation of the onboard equipment control model. The graphical model’s blocks represent different elements of the equipment, fields in the blocks are interfaces of physical devices, lines between the blocks are commutation connections, arrows are the directions of data transmission.

Figure 1. Graphical modeling of the onboard equipment command-and-measuring control (the following symbols are used: SC - spacecraft, OCS CU – onboard control complex, CCU – command and measuring system’s interface module, ODGS – onboard remote signaling equipment, GCC – ground control complex, TRANS – transmitter, RECV – receiver, HF – high frequency).

There are basic blocks in a graphical model that simulate the onboard systems: onboard control complex, onboard remote signaling equipment, the onboard digital computer complex, and also the elements of the onboard command-and-measuring system’s equipment: interface module, receiver
and transmitter. A ground control complex’s simulator is designed for command reception and
transmission and for the analysis of telemetry.

The model has a number of parameters and characteristics for setting different variants of the space
systems’ onboard equipment configuration. Software tools allow to create command bases of
arbitrary structures and set the methods of their generation, transmission and analysis. Each element
of the model can be graphically detailed. The level of detailing depends on the goals and tasks of
modeling. Figure 2 shows an example of graphical detailing of a transmitter model.

![Image of graphical detailing of the transmitter model.](image)

The detailing allows to study the features of the receiving-transmitting tract of the modeled system,
simulating transformation of the binary package into the signal transmitted to the onboard devices.
The model contains simulators of the coder, modulator, transformer and amplifier. The
decomposition of receiver model provides a full sequence of data transformation during transmission,
which allows to model evaluation of errors and losses at the tract due to different external influences.

Model’s elements’ functions can be performed in form of the condition-action rules, virtual tools
created in the Labview environment, or the C++ programs. There is a tight integration of graphical
tools and the simulation environment allowing to use graphical presentation during development and
conduction of simulation tests. Visual creation of a model with graphical elements provides
significant advantages due to simplicity and clarity of the onboard equipment design, analysis and
debugging of the model.

Intellectual Modeling Tools

At present, there are directions of development associated with usage of intellectual methods for
space equipment modeling. However the problem of creation of software modeling tools supporting
the SMP2 standard and providing transferability of models and their shared usage in big complex
projects is still topical.

Our intellectual modeling tools allow to model using rules of function of different elements of a
model and set the logics of their interaction in terms of the subject area. The rules are clear, so a
specialist can easily understand the structure of his own models as well as those created by other
specialists, and to see the modeled system in whole. Figure 3 is an example of the spacecraft
command-and-measuring system’s model. The model has a graphical presentation, a set of
parameters, a condition-action rules base and timers to logical inference or actions for the model’s
condition change.

The rules formally describe different variants of interaction of the graphical model’s elements for
simulation of the real equipment’s behavior. The knowledge base is structured in accordance with the
type of elements and the area of the rule’s action. There are common rules, elements’ function rules,
rules of command transmission to the spacecraft systems, rules inquiries of telemetry, rules of the
onboard equipment status control, equipment function mode and interfaces setting rules, etc. The
knowledge base is a storage of functional and technological processes allowing not only to observe, but also to control the logical inference.

Figure 3. Model description.

Intellectual modeling tools along with graphical tools provide high level of problem orientation of a simulation infrastructure. They allow a constructor to use familiar semantic structure for building models of the space systems’ onboard equipment function. Their implementation in a simulation infrastructure will increase the possibilities of model building and provide models’ transferability and integrability in complex solutions.

Our simulation infrastructure provides an onboard equipment designer with the ability to build space systems’ function models using the existing models of different manufacturers’ or creating his own ones in a special environment, not requiring additional programming skills from him. In order to build complex solutions and provide shared use of different models, we have developed software tools for simulation model integration.

**Simulation Models’ Integration Tools**

Space systems’ onboard equipment command-and-software control model consists of the models of different subsystems. Manufacturers’ specialization does not allow to create the whole complex of interacting subsystems. In our case realization of the models can be completed in form of the logical models, virtual tools or programs on C++. It is necessary to integrate the models developed by different manufacturers, in one solution. The complex model must provide study of the modeled object’s behavior in whole and influence of its parts on each other. At the same time, it must be easily modified and extended.

In order to integrate models, their different realizations are brought to a uniform type, which allows to use the same approaches and software tools of simulation tests’ preparation and completion. For all models, a metamodel is either created automatically or is downloaded during import. Metamodel is written in formal Simulation Model Definition Language (SMDL) [1]. In accordance with the SMP standard, the metamodel contains the structures of catalogue, configuration, assembly, planner, and the modeling methods’ realization in form of binary files. The information-graphical modeling tools allow to add the models with the command and telemetry structures, determine data reception-transmission interfaces, and to set commutation links and directions of informational interaction. Bringing of different models’ realizations to uniform components of the problem-oriented simulation environment allows an onboard equipment designer to build simulation models using different variants of their realizations. The user will be able to operate the model’s
functions equally, no matter whether they are realized in a special programming language or described in form of rules. Different realizations will be shown to him only in form of the graphical model elements’ icons (Figure 4).

![Graphical model elements presentation.](image)

In order to build a complex model of the onboard equipment command-and-software control, it is necessary to realize the models of the onboard system, command-and-measuring system and the reception-transmission devices. In Figure 4, the modeling methods of the onboard equipment simulator are realized in C++. Command-and-measuring system’s simulator modeling is completed on the basis of rules, and virtual tools are used to model the reception-transmission device. After they are integrated in the infrastructure, the user operates with graphical model independently from the way of their realization. Simulation is performed with the help of the SMP2 standard mechanisms: Time Keeper, Scheduler, Logger, Event Manager, Link Registry, Resolver. Such approach allows not only to make various and quality decisions, but also to exchange knowledge between the groups of the constructors with big experience in different aspects of the tasks under solution.

**Conclusion**

The simulation infrastructure created in accordance with the Simulation Model Portability standard and added with the information-graphical and intellectual modeling tools, is designed for the work of the specialists of different directions in the area of satellite systems’ equipment design.

Software modeling tools of the space systems’ onboard equipment command-and-software control are designed for technical projects’ preparation and analysis. The original methods and tools of the information-graphical modeling allow to build the models of space systems’ function and use them to transmit the knowledge about the onboard equipment’s structure. Setting of the model functions in form of rules will let an engineer with special knowledge about the work of the onboard systems, to easily build and then modify simulation models. The models integration method provides combination of different elements of the model in a one simulation complex.

Application of the onboard equipment command-and-software control modeling tools allows to increase the quality solutions at different stages of the space equipment's production lifecycle.

**Acknowledgement**

The reported study was funded by RFBR and Government of Krasnoyarsk Territory according to the research project № 16-41-242042.

**References**


