Rand Model Designer in Manufacturing Applications

D. Inikhov,* Y. Kolesov,**
Y. Senichenkov***

**MvSoft, Moscow, Russia, (e-mail: inihov@mail.ru)*
***R F C A, Moscow, Russia (e-mail: ybk2@mail.ru)*
**** St. Petersburg State Polytechnic University, St. Petersburg, Russia, (e-mail: senyb@dcn.ftk.spbstu.ru)*

Corresponding author-Yuri Senichenkov, (e-mail: senyb@dcn.ftk.spbstu.ru)

Abstract: Equation based computer modeling is used traditionally for investigation, simulation, and optimization a test subject at the early stage of designing. And with it a computer model may be a real time embedded component of a hardware-software system. Manufacturing models especially for hardware-software system are large-scale models designing by teams usually. Such complex projects demand using special tools for joint operation. Designing large-scale models in business competition at a stated time also needs using special tools for modeling and simulation such as the family of tools developed by MathWorks Inc. for example. In this paper we present a new tool named Rand Model Designer and demonstrate its properties and possibilities using examples of manufacturing models.

Keywords: modeling, simulation, virtual reality, simulators, training, real time computer systems, large-scale systems, mathematical models, state charts, hybrid systems, visual modeling languages.

1. INTRODUCTION

We selected two groups of manufacturing models for illustration Rand Model Designer's facilities. The models from the first group illustrate traditional using equation based computer models in industry for prototyping complex engineering systems at early development stage. The examples from the second group demonstrate using Rand Model Designer (RMD) for real time embedded applications.

A typical modern engineering system has hierarchical structure including a lot of different physical nature components functioning in various modes switched under events (Tiller M. (2001), Fritzson P. (2006)). RMD is based on object-oriented approach (Kolesov Yu. B. (2004)) to analysis and design of prototype of complex engineering system. RMD inherits object-oriented technology from UML (Rumbauth J., Jacobson I., Booch G. (2005)). and accommodates it for modeling multi-component dynamical and hybrid systems. Particularly it is possible by using slightly modified variant of UML state machine designing integrated model of complex engineering system switched between all allowable modes. Simulation in RMD also has its own particularity. Used numerical methods take in account the structure of solved current system and it allows increasing computation speed and cut down memory expenses. RMD's numerical method drivers use general-purpose methods by default but user can select more suited for his problem method from the RMD's numerical library.

2. MODELING TECHNOLOGIES OF RAND MODEL DESIGNER

Rand Model Designer (www.rand-service.com) is a tool from the family of the tools for modeling and simulation complex dynamical systems developed by MvStudium Group (www.mvstudium.com). RMD is visual equation based environment for object-oriented modeling of large scale real world or technical systems using special mathematical models called hybrid systems.

Hybrid systems are good for modeling real world objects with multiple-mode behavior (event-driven systems). We consider hybrid systems as generalization of classical continuous dynamical systems (Senichenkov Yu. B. (2004)). In hybrid systems local continues behaviors (modes) and discrete actions associated with them alternate one-by-one dependent upon control signals or taken place events. Discrete actions are used for modeling processes with negligibly small duration and they are interpreted as entry, exit, or transition actions for continues behavior. Visual notation for hybrid systems is called hybrid automation (Fig. 1). In RMD an extension of UML state machine named Behavior-Chart (Kolesov Yu. B., Senichenkov Yu. B. (2006)) is used. A local continuous behavior (extension of UML activity) of RMD's Behavior-Chart (B-Chart) is a solution of systems of algebraic-differential equations written in general form (unresolved respect to derivations, Fig. 2). In RMD any continuous behavior (formulas, substitutions, and equations) is written and edited in habitant mathematical form. Not sophisticated procedural language is used for description of discrete actions.

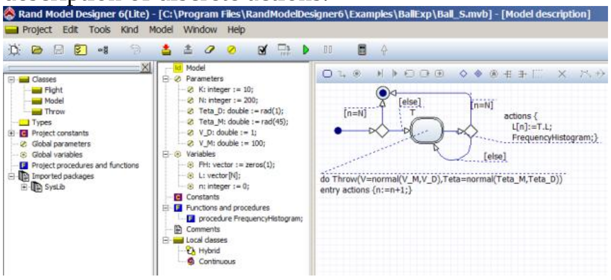

Fig. 1. RMD B-CHART with do-activity in the form of algebraic-differential equations.

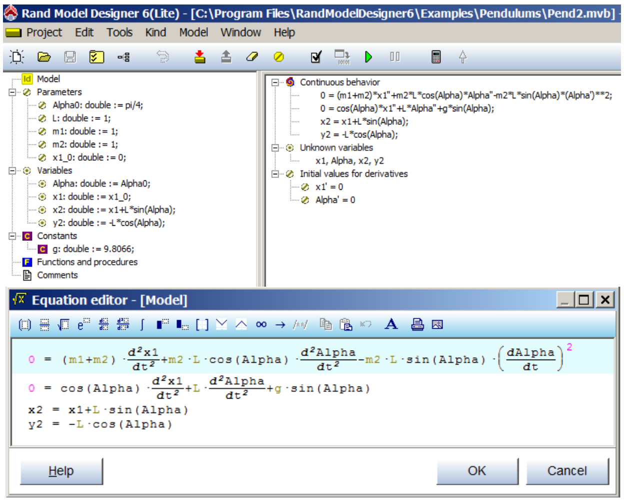

Fig. 2. RMD's Equation Editor

Model Vision Language (MVL) used in RMD maintains component modeling with oriented ("causal" modeling, Fig. 3)

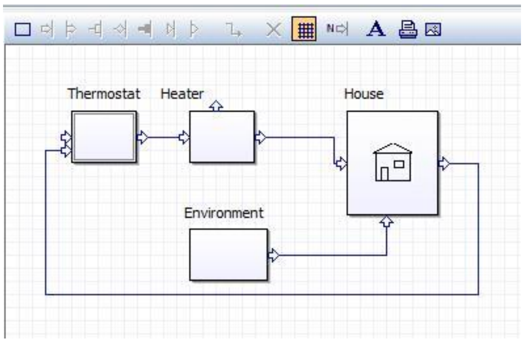

Fig. 3. "Causal" modeling block-diagram

and non-oriented ("physical" modeling, Fig. 4) components (Kolesov Yu. B., Senichenkov Yu. B. (2007)).
Any component may have its own "component" system of algebraic-differential equations or a set of such systems in case of hybrid systems. A set of components and their links (bonds) forms functional diagram of a whole model. Component and topological equations of a whole model form a final system of equations (total mathematical model). If there are hybrid automata into components then a current functional diagram and its current final system of equations depends on events. A current final system is constructed and transformed to form required by numerical method automatically on run time. Transformation of a system is needed and carried out for decreasing of number of unknowns, reducing index, changing structure of equations.

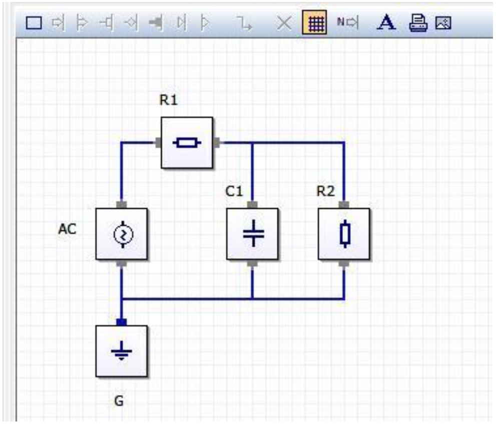

Fig. 4. "Physical" modeling block-diagram

RMD builds two sorts of executable models:

- A visual model that is Windows stand along application stands for testing, visual debugging, carrying out computational experiments, and processing results;
- A "hidden" model that is Windows dynamical linked library (dll) used as an embedded interactive application.

Hereby we can summarize:

1. RMD is oriented on well-defined mathematical models, which are "event-driven set of algebraic-differential equations". A dimension, type, and numerical properties of such systems depend on current model mode.
2. RMD's hierarchical component diagrams and B-Charts allow to user designing complex large-scale models in intuitive manner.
3. RMD's modelling language maintains technologies for causal and non-causal modelling simultaneously.

RMD is used for scientific research and teamwork computer-aided design of large-scale systems. In both cases object-oriented modelling renders assistance. Working out or library classes are used as building material for a complex model. Their instances may have parameters for customization, inherit properties of parents, and demonstrate polymorphous behaviour.
RMD has easy-to-use instruments for debugging, testing, and contrastive analysis of different model modifications as well.

3. MODELING-BASED DESIGN WITH RAND MODEL DESIGNER

Modeling-based design (MathWorks) based on executable specifications allows step by step complicating workable model and confronting model behavior with prescribed by specification using computational experiments. Specifications written in high level graphical modeling language are well-adapted for communication among developers. Necessity repeatedly carry out experiments obligates to have high level language for planning and carrying out computational experiments.

RMD is good for using modeling-based design. It has high level languages for modeling, testing, debugging, planning experiments. There are special visual instruments for parametric optimization, sensitivity analysis, and statistical experimentation.

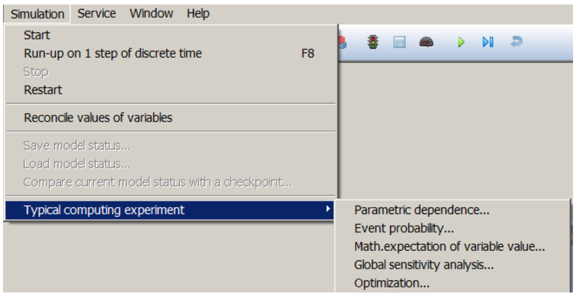

Fig. 5. RMD's tools for computing experiments

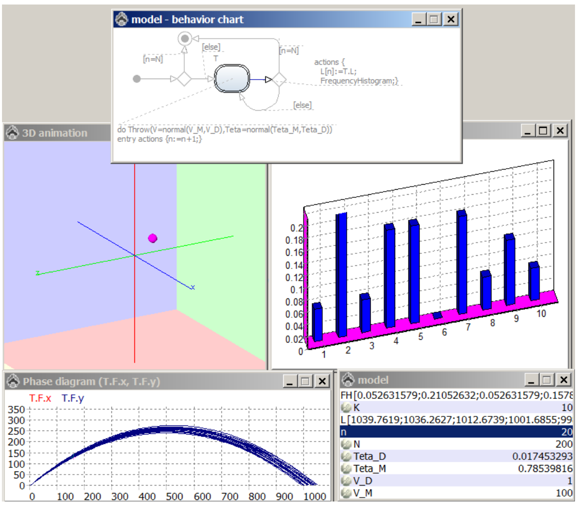

Fig. 6. RMD's statistical experiments

RMD's visual environment makes easier developing complex components by using "from the simple to the complex" technology. The first step is developing isolated model (isolated classic dynamical system). The second step is it's fitting out by hybrid behavior (hybrid system). The third step is transformation of isolated component to open one (component with external variables: "input", "output", "contact", "flow"). A transformation from "simple" to "complex" form may be done automatically.

RMD's graphical language maintains intuitive graphical forms for conventional description of component hierarchical systems used by engineers.

RMD's B-Charts are used not only as graphical specification of hybrid systems but as a component of visual language for planning computational experiments. If model is designed it is possible to consider it as a class of more complex model. This complex model may be described as hybrid system (plan of experiment) with local behaviors (objects of class with concrete parameters values).

4. RAND MODEL DESIGNER'S NUMERICAL SOFTWARE

RMD's numerical library contains different variants of numerical methods (Senichenkov Yu. B. (2004)). It allows taking in account specific forms of a model equations what usually leads to decreasing number of unknowns, increasing calculation accuracy, and acceleration of calculating speed. RMD has a set of instruments for analysing model numerical properties and comparison experimental results.

User-friendly mathematical notation for description local behaviors and B-charts complicate designing numerical library for modeling and simulation of component models especially for "physical" modeling. Total model B-chart generates a sequence of systems of algebraic-differential equations with different size, structure, and numerical properties on trajectory. Algebraic-differential equations in turn may take form of linear and non-linear algebraic equations, differential equations, high index algebraic-differential equations in the general form (Ascher U.M., Petzold L.R. (1998), Hindmarsh A.C. (1983), Li S., Petzold L. (1999)). However modern DAE Solvers are oriented on a single problem solved on a few consecutive intervals. All preliminary work should be crafted individually by specialists beforehand. It is strange but there is no universal numerical software intended for "hybrid" DAE. Modeling numerical libraries are formal integration of heterogeneous components usually without effective kernel and with numerous duplications.

Another problem is necessity of automatically analyzing every new system from job stream generated by model B-chart. Usually analysis has two stages that are symbolic or antecedent analysis, and numerical or posttest analysis. Symbolic (structural) analysis reveals systems that have no solutions for all possible values of parameters. It is checking up necessary conditions for solution existence. In the case of algebraic-differential equations sometime it is necessary to determinate and reduce index if possible. Numerical method checks up sufficient conditions on run time. Run time analysis implies numerical solution accuracy ranking for algebraic systems using condition numbers, stiffness ranking for differential equations and so on. It is important to underline that it is necessary to ensure the optimal calculations on whole trajectory for event-driven systems but not for separate problem. Just the same problem we have for computational experiment, testing, and parametric optimization when it is important to minimize complex experiment total time.

There is special block named Analyzer in RMD for system properties analysis. For "physical" models with hybrid behavior symbolic analysis should be executed on run time so we need fast algorithms for it.

5. BALLISTIC MISSILE FLIGHT

Going into orbit and flight of free stage ballistic missile (Benevelsky C.B., Kolesov Yu. B. (2009)) is an example of traditional using of modeling multimode device (*Fig. 7*).

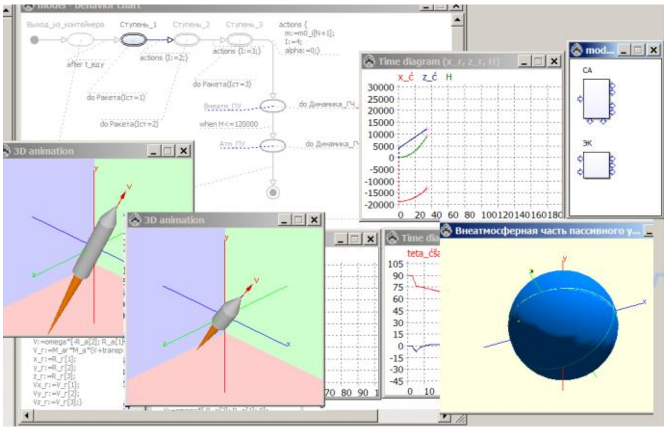

Fig. 7. Free stage ballistic rocket flight: going into orbit

States of the main B-chart describe boost phase (1-st, 2-nd, 3-thd rocket stage running), midcourse stage, and terminal stage. Stage changes is associated with changing parameters values or/and change of system of equations of motion. The main B-Chart is hierarchical: each of stage states has hybrid behavior and its own B-Chart in turn. Nested B-Charts supports different control modes of stage stop (cutoff of thrust rocket engine command, fuel burn-out). Using Object-Oriented-Modeling (OOM) the complex model behavior is divided on rocket motion, solid-propellant rocket engine dynamics, and control and each aspect is represented by its own class. This decomposition makes model intuitive and simplifies testing and debugging. In addition OOM allows reusing of debugged models. In our case the models of International Standard Atmosphere and Gravitational Field Model for the Earth are imported from previously worked out libraries.

6. COMPLEX SIMULATOR DESING

Using RMD system for complex simulators design at "Transas Technology" (www.transas.com) is a good example of industrial second type modeling (Kiptiliy D. B., Kolesov Yu. B., Lebedev D. V., Senichenkov Yu. B., Tarasov S. V. (2011)).

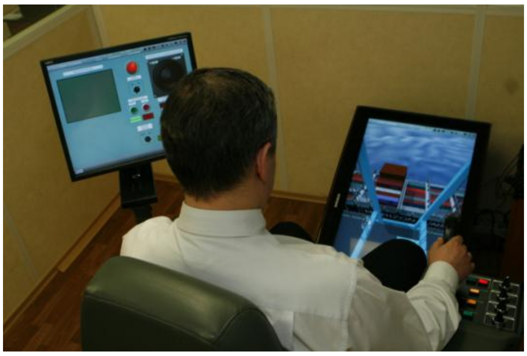

Fig. 8. Simulator operating position.

Modeled technical objects (such as mechanical, electrical, hydraulic, pneumatic, etc.) are described by systems of nonlinear differential-algebraic equations with a dimension of up to ten thousand equations and variable structure with complex switching logic. Obviously, to create such a model manually even at the mathematical level is almost impossible. The only way is to create applied class libraries, where the typical components for the specified field classes with inheritance hierarchy are defined. Then the model is combined from the components – copies of the classes – connected together (non-directly in general). The behavior of the model is reconstructed automatically. Object-oriented capabilities of RMD, supporting creation classes of active dynamic objects and data types, inheritance and redefinition of classes, independent design and import of packages by users, proved to be enough for simulators design. Using UML state charts instead of specific Modelica language constructions also proved to be a very successful. The most difficult requirement was the real-time working of automatically generated built-in modules. To provide this opportunity the numerical methods and assembly, analyzing and total system of equations convert (using extended UML state charts for hybrid systems behavior description requires this actions at running) should be conducted fast enough (within a tenth of a second). To a large extent, these requirements are contradictory: the experience has shown, that to provide the high speed of numerical methods it is necessary to conduct symbol convert to simplify the system and to define the special structures, independent blocks and etc. in it, while the time of such analysis and converts is quite limited. As a result, an acceptable compromise was found.

Fig. 9. Visual presentation of wharf crane.

The experience revealed some important problems while debug component models of high complexity. There are some situations, when the particular connection of standard components leads to the incorrect situation, that is not provided by the developers of the class library: at the specific points the system of equation becomes structurally flawed (underdetermined, over determined or degenerate) or numerically degenerate. To define which system components caused the situation is quite difficult if the system has high dimension. Due to this reason, the debugging tools were supplemented with the capability to visualize Jacoby matrix, its eigenvalues for the current system of equations and structure matrix of current total system of equations that provide helping information for the model developers. The capability of interactive step by step debug of discrete actions is vital for mining logic switching.

7. STRATEGIC AUDIT

Let us consider another hardware-software system named «Strategy - Chamber of Accounts» developed by St. Petersburg Institute for Informatics and Automation of RAS. The system is intended for strategic audit of socio-economic development of Russia Federation. In this complex selection of development strategy for the county or a region is based on results of simulation. During simulation stochastic experiment with a model of stock of projects and measures planed for selected country or region macroeconomic goals achievement is carried out. On the one hand it is impossible to build a model of stock of projects and measures by hands because all needed information for concrete project is stored in huge project management database («Spider Project») and users are non-specialist neither in programming nor in modeling. In the meanwhile a macroeconomic model may be developed a priori by hands and customized with the help of parameters for concrete region. The solution of this problem was found with the help of OOM. A macroeconomic model is designing, debugging, and testing in RMD as a class of «Macroeconomic» Package. For modeling a stock of projects and measures was developed special classes that are “Job”, “Computer experiment” and so on. For the integration software complex «UPE&PlanDesigner» (developed by «SoftProm») it was designed special tool which is able transform data base specification of model of a stock of projects and measures to build corresponding model into RMD modeling language automatically using special applied library classes.

Fig. 10 illustrates this technology. The shown model is the object of “Macroeconomic” class. Class “Macroeconomic” uses inner classes RMD specification generated automatically using data base specification. RMD builds embedded executable model, runs it, and exports results for visualizing in «UPE&PlanDesigner».

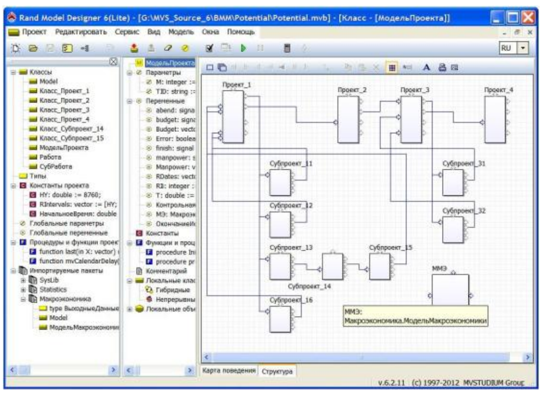

Fig. 10. RMD’s macroeconomic model.

7. USING RMD IN EDUCATION

Virtual laboratories for schools and university are special type of simulators (Fig. 11).

In the article (Biryukov S.V., Guskov D.N., Fedyanin V.V., (2005)) prof. Biryukov S.V wrote: “One of the most difficult problems in teaching physics is real-world experimental skills training. Real world experiments are often used, but it is too expensive. Computer simulations can be helpful in many cases instead. Computer model must be cheap, variable, and must present initial example to make pupil's own model. Menu, Help, Tutorial, and Teaching materials must be in student's native language. All these requirements are satisfied by Model Vision Studium (Former name for RMD).”

Experiments with the given models and making mathematical and then computer models of the real world phenomena are good tools for student's understanding assessment. Physical phenomena simulation is important in the main physics education consequence: theory – solving theoretical and applied problems – virtual and real-world experiments – assessment.

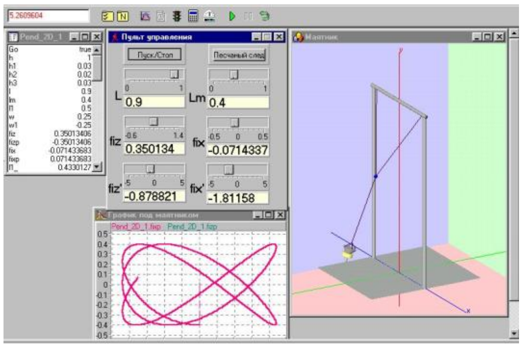

Fig. 11. Professor *Biryukov’s S.V. model – test bench for schools.*

Several virtual workshops in the course of General Physics (Mechanics) developed in Model Vision Studium simulator are presented in (Biryukov S.V., Guskov D.N., Fedyanin V.V., (2005)): “Motion investigation using Atwood's machine”, “Bullet velocity determination by kinematic and dynamic methods”, “Measuring moment of inertia of a bicycle's wheel using dynamic and oscillation methods”, etc.

Prof. Biryukov S.V consider: “These workshops are very similar to real world workshops in the General Physics Labs of the Moscow Pedagogical State University and they are used as an additional home study for campus students. Furthermore it can be used in Physics distance leaning as initially it was developed for open education portal. “Physical pendulum oscillations” is a workshop with real-time simulation and at the same time simple real world experiment with hand-made physical pendulum – modified additional computer mouse. Modification is so simple that it can be done by many students and thus it can be recommended as real-world and real-time computer experiment in distance learning or just for fun. "

8. MODEL CONVERTING

There are a lot of tools for modeling and simulation of complex dynamical systems (Breitenecker F., Proper N. (2009)). Sometimes it is difficult to replace habitual tool for new one especially if old one is used as component of industrial technology. One of the reasons is necessity rewriting applied libraries. To overcome this typical encumbrance is possible using convertors (preprocessors) from one modeling language to another.

RMD has special library (SysLib) which contains Simulink-like components (Fig. 12-13). They were written by hands and intended for Simulink's users doing their first steps in RMD.

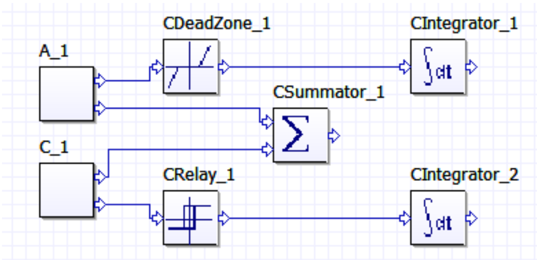

Fig. 12. RMD's Simulink simulator.

Practice shows that convertor from Simulink to RMD models (and vise-versa) is required not only for applied libraries but for comparative study of separate models. Additionally model converting allows using for investigation tools missing in the origin environment.

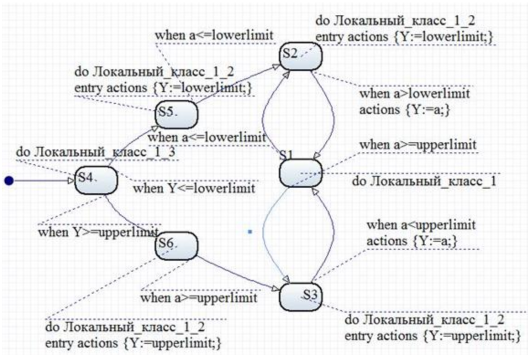

Fig. 13. RMD's "Integrator" simulator.

RMD convertor transforms Simulink specification of a model to MVL language using extended SySLib library for Simulink blocks.

REFERENCES

Ascher U.M., Petzold L.R. (1998). *Computer methods for ordinary differential equations and differential-algebraic equations, SIAM.*

Benevelsky C.B., Kolesov Yu. B. (2009) *Modeling. Object-Oriented Modeling for exterior ballistics problems.* St. Petersburg, STU, 126 p.

Breitenecker F., Proper N. (2009). Classification and evaluation of features in advanced simulators. Proceedings MATHMOD 09 Vienna, Full papers CD Volume.

Biryukov S.V., Guskov D.N., Fedyanin V.V. Visual Simulation of Physical Processes in Model Vision Studium, (2005). Proc. of the Int. Conf. "Informational Technology in Education", Moscow

Brown P.N., Hindmarsh A.G., Petzold L.R. (1993). Using Krylov methods in solution of large-scale differential-algebraic systems. Technical report. Numerical Mathematics Group, UCRL-JC-113507 (https://computation.llnl.gov/casc/nsde/pubs/221215.pdf)

Fritzson P. (2006). *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. Wiley-IEEE Press.

Hindmarsh A.C., Peter N. B., Petzold L.R. (1993) Using Krylov Methods in the Solution of Large-Scale Differential-Algebraic Systems.

Hindmarsh A.C. (1983), ODEPACK. A Systematized Collection of ODE Solvers in Scientific Computing, R. S. Stepleman, editor.

Kiptiliy D. B., Kolesov Yu. B., Lebedev D. V., Senichenkov Yu. B., Tarasov S. V. (2011) Object-oriented approach to designing cargo shipping operations simulator. Proceedings of SCM MEMTS, Saint-Petersburg.

Kolesov Yu. B. (2004) *Object-Oriented modeling of complex dynamical systems*. St. Petersburg, STU, 238 p.

Kolesov Yu. B., Senichenkov Yu. B. (2006) *Modeling of Systems. Dynamical and Hybrid Systems*. St. Petersburg, BHV, 224 p.

Kolesov Yu. B., Senichenkov Yu. B. (2006) *Modeling of Systems. Object-Oriented Modeling*. St. Petersburg, BHV, 186 p.

Kolesov Yu. B., Senichenkov Yu. B. (2007) *Modeling of Systems. Practical Work on Computer Modeling*. St. Petersburg, BHV, 352 p.

Li S., Petzold L. (1999) Design of new DASPK for sensitivity analysis. S. Technical Report, Department of Computer Science, University of California Santa Barbara, (www.cs.ucsb.edu/~cse/Files/LiPetz99.ps)

Rumbauth J., Jacobson I., Booch G. (2005). *The unified modeling language. Reference manual.* Second edition. Addison-Wesley.

Senichenkov Yu. B. (2004) *Numerical Modeling of Hybrid Systems. Practical Work on Computer Modeling*. St. Petersburg, STU206 p.

Tiller M. (2001). *Introduction to physical modeling with Modelica.* The Springer International Series in Engineering and Computer Science.