Physical Modeling of Hybrid Systems
with Rand Model Designer

D. Inihov,* 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,
Russia, (e-mail: senyb@dcn.flk.spbstu.ru)

Abstract: New visual environment Rand Model Designer developed by MvStudium Group for modeling and simulation of complex dynamical system is presented. Modeling language of the tool is based on object-oriented approach and maintains component modeling with oriented ("causal" modeling) and non-oriented ("physical" modeling) components with hybrid (event-driven) behavior. Problems of performing current final systems of algebraic-differential equations for "physical" hybrid models, using structural analysis for modeling (identification of «unsolvable» systems) and simulation (computation speed) are considered.

Keywords: complex dynamical systems, hybrid systems, visual modeling languages, numerical software, structural analysis, differential-algebraic equations, direct and iterative methods for solving large systems of linear algebraic equations.

1. INTRODUCTION

Modern tools for modeling and simulation of complex dynamical systems are based on object-oriented modeling and maintain component modeling with oriented ("causal" modeling) and non-oriented ("physical" modeling) components.

Using state machine as a specification of event-driven component behaviour makes changes in algorithm of building current final system of algebraic-differential equations for "physical" modeling in comparison with "causal" modeling. For "physical" modeling this algorithm has greater time complexity. In spite of this we suggest performing a current final system during execution. It allows modeling hybrid systems without any limitation on type and structure of a local system of algebraic-differential equations.

For embedded systems it is very important providing breakneck computation speed. A possible way of decreasing model run-time is updating numerical methods. One of the basic auxiliary operations of numerical methods for solving algebraic-differential equations is solving linear system of algebraic equations. Modern solvers of algebraic-differential equations based on direct methods for solving linear algebraic systems of equations deal with general, band and sparse matrices. The list of practically important equation structures is bigger. It is possible to modify numerical methods for solving algebraic-differential equations taking in account different types of equation structure.

Modern industrial projects require special technology for debugging and testing. Jumbo size of whole model hybrid automation allows analysing separate trajectories only.

Therefor information about separate trajectory should be manifold. It may be a list of visited states of total hybrid automation on trajectory and sequence of final systems, type and structure of final systems, Jacoby matrices values, sequence and results of discrete actions, etc.

Rand Model Designer (www.rand-service.com) developed by MvStudium Group (www.mvstudium.com) is industrial visual environment for designing complex dynamical systems. In comparison with Model Vision Studium it maintains "causal" modeling event-driven systems, suitable for designing embedded systems, allows taking in count structure of solved systems, and has powerful instruments for testing and debugging large-scale models.

2. RAND MODEL DESIGNER

Rand Model Designer (for short RMD - www.rand-service.com) is a commercial version of Model Vision Studium (Breitenecker, 2009) (www.mvstudium.com). RMD is an equation based graphical environment for object-oriented modeling and simulation of complex real world or technical systems. It is adjusted to special mathematical models called hybrid systems.

Hybrid systems are the best for modeling multiple-mode or event-driven systems. So RMD’s model component usually contains hybrid system. A visual form of hybrid system is called hybrid automation. RMD’s hybrid automation (Fig. 1-2) is an extended UML state-machine and called Behaviour-Chart (B-Chart).
A continuous behaviour in RMD B-Hart is described by algebraic-differential equations written in usual (symbolic) mathematical form. Discrete behaviour or algorithms are written in the tool modeling language named Model Vision Language (MVL).

MVL supports component modeling with oriented and not-oriented components. A RMD component is an open hybrid system. Open hybrid system has a set of special different kind variables (oriented: inputs, outputs, non-oriented: contacts, flows) that may be used in coupling equations. A RMD’s component diagram is a set of components and couples between them. A final system of equations to be solved for simulation contains component equations and coupling equations. B-Chart generates a sequence of current final systems of equations on a trajectory. RMD forms, analyses and reduces a current final system on run-time. “To form, analyse and reduce” means to build automatically equations set using B-chart, analyse its structure, decrease its dimension, reduce them in compliance with chosen for solution numerical or symbolic method.

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), which may be used as an embedded interactive application.

RMD has its own distinguishing characteristics:

1. It 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. Hierarchical component diagrams and B-Charts allow to user design complex large-scale models.
3. MVL maintains technologies for causal and non-causal modelling simultaneously.

RMD is used for scientific research and teamwork computer-aided design of large-scale systems (Fig. 3). In both cases object-oriented modeling 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.

RMD numerical library contains different variants of numerical methods. It allows taking in account specific forms of a model equations what usually leads to increasing calculation accuracy and acceleration of calculating speed. RMD has a special visual language for planning computational experiments, a set of instruments for analysing model numerical properties and comparison experimental results.

RMD has easy-to-use instruments for debugging, testing, and contrastive analysis of different model modifications as well.
3. PHYSICAL MODELING

It is possible to build multi-component “physical” models using a lot of tools (Breitenecker, 2009). Most of them have data type “Record” in their modeling languages used for constructing «Connectors». User’s defined «Connectors» make “physical” models expressive and really physical. Models with complex «Connectors» inherit not only real word object structure but and its bond nature.

![Image](image1.png)

Fig. 4. “Physical” modeling with RMD: an electrical circuit.

Designing RMD we have based on Modelica experience in the field of “physical” modeling (Fritzson, 2006; Tiller, 2001) and our own MvStudium’s way of modeling hybrid systems. Used in RMD B-Chart is UML state machine (Rumbauth, 2005) without orthogonal and history states specifically.

![Image](image2.png)

Fig. 5. “Physical” modeling with RMD: B-Chart for an “Jumper”.

UML reference is no coincidence here. We want to underline that RMD’s version of Model Vision Language (MVL) is an UML based language. We consider UML as appropriate standard for object-oriented modeling of complex (multi-component, large-scale, event-driven) dynamical systems. Modern tools for modeling and simulation complex systems obviously use special forms of class, object, component, structure, state machine diagrams, and other UML graphical elements in their modeling languages. Active classes and active objects play special role in describing component’s continuous behaviour. Continuous behaviour of complex dynamical systems (piecewise continuous solutions of algebraic-differential equations) may be considered as extension of UML activity. Passive classes are used for discrete actions as usual. Thereby UML semantic may be used as a standard for modeling complex dynamical systems.

![Image](image3.png)

Fig. 6. “Physical” modeling with RMD: run-time visual representation of the electrical circuit specification.

“Physical” modeling for components with B-Charts has its own characteristics. It is well known that time of building current final system depends on types of components. Solvability of any possible current final system of equations generated by B-Chart for oriented components (blocks in Simulink notation) may be checked while compiling. It is impossible to do so for oriented components with hybrid behaviour in general case. Component equations for oriented components are usually underdetermined. We are obliged analysing solvability of current configuration of component equations on run-time period if we want examine local behaviour of hybrid system as an arbitrary dynamical system.

Component’s equations in RMD

\[ f_i \left( \frac{ds_i}{dt}, s_i, y_i, t \right) = 0, i = 1, N; \]

were N is number of components, respect to vectors of component state variables \( S_i \) and vectors of component external variables \( Y_i \) together with coupling equations form a current final system

\[ F_i \left( \frac{dx_i}{dt}, x_i, t \right) = 0. \]

Type and structure of final system are analysed, and it transforms to semi-explicit form of algebraic-differential equations

\[
\begin{align*}
\frac{ds}{dt} &= z \\
F(z, x, t) &= 0 \\
\end{align*}
\]

if necessary.
4. STRUCTURAL ANALYSIS

In RMD structural matrix is used for building a current final system. Lines of structural matrix correspond to components equations. Colons correspond to current model variables. Coefficients of structural matrix have {0, 1} values. Value “one” in the position of colon and line crossing says that an appropriate variable is used in equation under consideration. A system be called structural nonsingular (formally solvable) if it has square structural matrix with full transversal (all diagonal elements are equal to “one”). RMD’s Analyser inspects the current structural matrix for full transversal, and tries to reorder colons and lines for building matrix with full transversal if necessary (mode “realize structural analysis”). If reordering is impossible Analyser tries determining problem by itself. In this case user gets a warning and the list the variables obtained forced values.

Tabular procedure of building current final system of equations based on structural matrix is used in RMD. Non-linear component equations are complemented by topological equations, that are systems of linear algebraic equations with {0, +1, -1} coefficients. Analyser reorders equations if necessary. Special structure of topological equations allows decreasing dimension of a final system. RMD’s Analyser first of all divides a final system on non-linear and linear part and analyse their specific where there is much to gain from it.

5. STRUCTURE OF SOLVED SYSTEM

Modern numerical methods for solving algebraic-differential equations (Ascher, 1998; Hairer 1996) are based on auxiliary operations: calculating Jacoby matrix and solving systems of linear algebraic equations. Calculation of Jacoby matrix and solving systems of linear algebraic equations may be done numerically or symbolically. Numerical methods for solving systems of linear algebraic equations may be direct or iterative.

Preconditioned Krylov methods for solving linear algebraic systems are used in DASPK (Brown, 1998; Li, 1999). A rate of convergence of preconditioned Krylov methods depends on preconditioner matrix. A regular way of building preconditioner matrix automatically for component models is unknown now. We are doing our first steps in this direction.

Direct methods for solving large systems of linear algebraic equations try taking into account matrix structure. RMD’s methods for solving large systems of linear algebraic equations distinguish band, block-diagonal, block-triangular, sparse and general matrices (Fig. 6-9).
Matrix structure may be known beforehand or it is possible determining it automatically. It is permissible to try reordering matrix colons and lines for transforming matrix to matrix with wanted structure (Pissanetzky, 1984). All this should be used for increasing computation speed.

RMD’s user can inform Analyser about a final system structure with the help of special operators. For example user can place operator «block name… end block» among component equations (Fig. 10, fragment “block NLB”) and Analyser would inspect if final system composed from such blocks has block-diagonal structure or it hasn’t.

$$A \mathbf{w}_j = \mathbf{b}_j, j = 1,m$$

where A is matrix with constant coefficients (see Fig. 10, fragment “linear block VLB” and take in account that X,Y,Z are m-dimensional vectors and “aij” are constants).

5. DEBUGING AND TESTING

Behaviour of a hybrid system is a set of trajectories generated by hybrid automation in state variables’ domain of definition. Final (aggregate) hybrid automation for a large-scale multi-component system is beyond RMD’s user. Thereby user can debug and test separate trajectories only. There are three levels of debugging in RMD. User can get information about trajectory for given initial conditions as the sequence of visited states of component’s hybrid automata with names of activated transitions. In this case for each visited state of final hybrid automation short description of current final system is available: type (algebraic equations, differential equations, algebraic-differential equations), initial vector (plus a vector of consistent initial conditions for algebraic-differential equations) and the solution in final point. If you need more information about continuous behaviour you may request output of a procedure for equation’s residual calculation called by numerical method and intermediate values of the solution on interval. All this information may be associated with internal variables used for calculations by numerical method and with user defined names of variables.

All debugging information is saved as a text file. Additionally user can get some information about a model in graphical form during model execution. First of all it is possible to watch and print current final system in mathematical form. There are special windows for Jacoby matrix and its eigenvalues onto solution.

6. CONCLUSIONS

Rand Model Designer (under the name MvStudium Professional) is used now for designing shipping cargo operations simulators (Kiptily, 2011). Sea cargo operations simulators are hierarchical multi-component systems with oriented and non-oriented components. Theirs training purpose assumes using hybrid systems with a lot of hierarchical states and local behaviors as mathematical model for supporting active learning process (play behavior). Program realization of mathematical model has to run very fast (it should take no more than 10% of simulation total time in our case study). Rand Model Designer object-oriented approach ensures designing large-scale models. The Structural Analyzer allowed not passing “unsolvable” systems to program realization of numerical methods and raised calculating speed.

REFERENCES


