A new tool for visual modeling - Rand Model Designer 7.

Andrey A. Isakov *, Yury B. Kolesov *,
Yury B. Senichenkov*

**National Research University St. Petersburg State Polytechnical University, Russia, 195251, St. Petersburg, Polytechnicheskaya, 29 (Tel: +7 (812) 297-1616; e-mail: senyb@ dcn.icc.spbstu.ru).*

Abstract: MvStudium research group has announced the new version of visual environment for modeling and simulation of complex dynamical systems Rand Model Designer 7. It has two principal differences in comparison with previous versions: a) now it is possible using dynamic objects of components with «input-output» or «contact-flow» external variables for solving problems of queueing theory («agent-based» approach); b) global system of equations for a local behavior of component model with hybrid behavior is always built, analyzed and transformed on run time.

Keywords: modeling and simulation, visual environment, complex dynamical systems, hybrid systems, component models, automatic building and transforming equations, algebraic-differential equations, structural analysis, block triangular form.

MvStudiumGroup research group supports three visual tools for modeling and simulation complex dynamical systems: Rand Model Designer (www.rand-service.com, www.mvstudium.com) for common using, TransasProf (www.transas.com) for designing real time marine training simulators, and open-source tool OpenMVLShell (https://dcn.ftk.spbstu.ru/) for education (Isakov and Senichenkov (2010, 2011)). All of them use object-oriented modeling language Model Vision Language (Kolesov and Senichenkov (2007)). Model Vision Language (MVL), likewise Modelica language (Fritzson (2011)), oriented on modeling hierarchical event-driven component systems. It is based on Unified Modeling Language's de facto standards, accommodating them for hybrid systems. Hybrid systems are considered as an extension of classical dynamical systems (Senichenkov (2004), Kolesov and Senichenkov (2006), Kolesov and Senichenkov (2012 ,2014).

The main concept of MVL is «active dynamical object» with global behavior described by Behavior-Chart (modification of UML's State Machine) and local behaviors in the form of algebraic-differential equations (Kolesov and Senichenkov (2007)).

Building global systems of equations for composition of component's automata is the main difficulty for component models with hybrid behavior. When number of components and number component's hybrid automata states increase, the number of possible states for composition became extraordinarily large. Alternative for building all possible systems beforehand is building only realized systems directly on run time. In large scale event-driven systems the number of realized local behaviors is large too. It is very important to know the structure of solving system. Systems with block triangular structure are abundant problems in practice. Using numerical methods taking block triangular structure in account increases speed of computer modeling. The problem of automatic building global equations for local behavior of component model with hybrid behavior and automatic detecting or transforming its structure on run time is considered.

The simplest RMD-7 user's equation form is a form for describing local behavior of isolated hybrid system. It is a system of differential equations with substitutions (1):

$$w = Subst(w,C),\ F(\frac{ds}{dt},s,w,t,C)=0,\ Out(y,s,w,C)=0,$$

$$w\in\Re^{k1};\ C\in\Re^{k2};\ F,\frac{ds}{dt},\ s\in\Re^{n};\ Out,y\in\Re^{m} \qquad (1)$$

It contains algebraic-differential equations $F(\frac{ds}{dt},s,w,t,c)=0$, respect to state variables $s\in\Re^n$, with constancies C and substitutions $w=Subst(w,C)$, and equations $Out(y,s,w,C,t)=0$ respect to output variables $y\in\Re^m$. Systems of equations in form $F(\frac{ds}{dt},s,w,t)=0$ will be automatically transformed with the help of Gear's substitution to

$$\left\{\frac{ds}{dt}=z;\quad F(z,s,w,t)=0, s(0)=s_0\right. \qquad (2).$$

Special cases (linear and non-linear variants) of form (2) are

$$\begin{cases}\frac{ds}{dt}=F(s,w,t)\\ G(y,s,w,t)=0, s(0)=s_0\end{cases} \qquad 2.a$$

$$\left\{\frac{ds}{dt}=F(s,w,t), s(0)=s_0,\right. \qquad 2.b$$

$$\{G(s,w,t)=0 \qquad 2.c$$

The linear form is recognized automatically.

There are special Solvers for each type of equations. The special role plays Solvers for Non-linear Algebraic Equations (NAE). To solve NAE is necessary in implicit methods for Ordinary Differential Equations (ODE), in methods for finding consistent initial conditions and solving Algebraic-Differential Equations (DAE). NAE are solved with the help of Newton method, and in case of its failure, Powell method is used. Solving of linear systems of algebraic equations (SLAE) is the main operation for Newton method in turn. Newton method may choose appropriate modification of Gauss method with taking structure of solved system matrix in account: dense, band, sparse, block triangular.

Any techniques decreasing computational costs of numerical solution are very important. Computing matrix block triangular form if possible is well known and commonly used trick (Duff (1977)). In RMD-7 this technique is used for structural matrix of equations. Structural matrix of a system is a matrix with {0,1} elements pointing out occurrence of unknowns in equations. RMD computes block triangular form for structural matrix of any allowed types of systems (SLAE, NAE, ODE, DAE) with the help of Tarjan algorithm and solves only subsystems for strongly connected blocks.

Tarjan algorithm in RMD is used for reordering substitutions to computable sequence of formulas and detecting equations («algebraic loops») among them too.

Building and transformation global equations cause maximum difficulties for component models with «contacts-flow» external variables («acausal blocks»). It is well known that if even local behaviors of components' automata do not contain high-index DAE, they can appear in their composition. So it is necessary to analyze all states of composition, if we want detect and build all high-index DAE systems beforehand.

RMD's Analyzer detects high-index DAE on run time and builds new additional equation for numerical differentiation.

The structure of RND's Numerical Library and algorithm of interaction between Numerical library and Model Engine have been changed for implementing new approach to building global equations on run time.

If any new event has occurred, the control program estimates necessity of rebuilding of current solved system. New system is built if it is necessary.

If the current global system has full transversal:

Tarjan algorithm is used for building block triangular form of structural matrix.

Strongly connected components (diagonal blocks of reordered structural matrix and associated with them systems of equations) are analyzing for choosing appropriate Solvers.

Solving a system of equations corresponding to diagonal block with the help of suitable Solver will be named «subtask».

Next step is building the condensation of the Tarjan's algorithm graph. The condensation is used for construction subtask queue. Subtasks may be executed sequentially or parallel. The information about computer hardware (number of processors, number of kernels for a processor) needed for creation a thread pool for parallel execution is determined automatically. Threads are loaded by subtasks which are ready for execution. The subtask readiness for execution determinates using condensation. Initially all nodes (subtasks) of the condensation are marked as «Unresolved». If an «unresolved» node has no input edges or input edges start from the nodes with solved systems then control program changes its status for «Ready to start», otherwise it will have status «Not ready to start». Solving subtask's system has name «Solving», after ending of solving it becomes «Solved». The calculation comes to an end when all nodes become «Solved».

The new approach with block triangular form, different Solvers for each subtask, and threads was compared with old one. For comparison was used a set of models developed by Transas company (http://www.transas.com/products). The results of numerical experiment for most difficult problem are shown in Table 1. The computer used for calculations had four processors, so it was possible to create maximum four threads, but even if only one thread was used then total time of calculations decreased in two times.

Table 1. Product Tanker, Cargo System, about 2500 equations.

Thread Count	1	2	4
Old Numerical Library	8,53	-	-
New Numerical Library	4,72	3,80	3,67
Time difference	**44,67%**	**55,45%**	**56,98%**
Time on Right Part Calculation	2,44	2,47	2,47
Right Part Calculation Count	28964	28964	28964
Control Time	2,28	1,33	1,20
From one thread	100,00%	**58,33%**	**52,63%**

REFERENCES

Duff I.S. (1977). MA28—A set of Fortran subroutines for sparse unsymmetric linear equations. Technical Report AERE R8730 (SR-2040 8.0 edition) HMSO, London

Isakov A.A., Senichenkov Yu. B. (2010). OpenMVL is a tool for modelling and simulation. //Computing, measuring, and control systems, pp. 84-89, SPBSPU, St. Petersburg.

Isakov A.A., Senichenkov Yu. B. (2011). Program Complex OpenMVL for Modeling Complex Dynamical Systems // «Differential equations and control processes» Journal, Mathematics and Mechanics Faculty of Saint-Petersburg State University, St. Petersburg.

Kolesov Yu.B., Senichenkov Yu.B. (2006) *Modelling of systems. Dynamical and Hybrid systems.* 224 pp, BHV, St. Petersburg.

Kolesov Yu.B., Senichenkov Yu.B. (2007) *Modelling of systems. Object-oriented approach*. 256 pp, BHV, St. Petersburg.

Kolesov Yu.B., Senichenkov Yu.B. (2012) *Mathematical Modelling of component systems.* 223 pp, SPbSTU, St. Petersburg.

Kolesov Yu.B., Senichenkov Yu.B. (2014) *Mathematical Modelling of hybrid systems.* 236 pp, SPbSTU, St. Petersburg.

Pissanetsky S. (1984). *Sparse matrix technology*, Academmic Press, London.

Senichenkov Yu. B. (2004) *Numerical modelling of hybrid systems.206 pp, SPbSTU, St. Petersburg.*

Fritzson P. (2011) *Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica,* 232 pp, Wiley-IEEE Press.