УДК 004.052; 519.62
P. Serdiuk, O. Tomjak
Lviv Polytechnic National University
Software development chair

SOFTWARE ARCHITECTURE FOR VERIFICATION AND AUTOMATED TESTING OF PROBLEM SOLVING METHODS FOR MATHEMATICAL PHYSICS

The article deals with software architecture for testing and verification methods for solving systems of equations in partial derivatives describing physical processes. Developed automated generation of processor architecture problems, their solution and comparative analysis with other methods of analysis tasks mathematical physics. Analyzed with software on shell COMSOL 4.2.

Key words – automated testing, differential equations, mathematical physics, verification.

Introduction

There's important aspect of development of new methods to solve partial differential equations - their verification and testing. One of most complex problems of automated testing is coverage of various models by test cases. In this paper we suggest automatic generation of test models for verification of mathematical methods to solve problems of mathematical physics.

Sequential scheme of package generation to produce testing templates of sequential chain is used to verify and assure convergence of methods of solving problems of mathematical physics. Such technique considerably increases the number of possible target problems and reduces excessive need to specify state variables, that sometimes appears during usage of standard algorithm[1, 2].

Today many methods to solve partial differential equations of mathematical physics and initial boundary value problems for linear and non-linear integral equations with partial derivatives[3,4] In this paper an empirical approach is used to measure solving method characteristics, which are used for verification tasks, convergence analysis and solution stability analysis. This empirical approach in essence is automatic generation of input data for solving methods and their further analysis. Software architecture to implement this method is also covered in this paper. Software consists of translators to mathematical scripting languages, generator of test constructs and verification module.

The task of automatic generation test constructs means to produce input data with corresponding constraints: spatial, boundaries for physical values and constructs. Generation of complex constructs and their processing is covered in [5], where authors suggest method of processing surfaces, that are generated as PDE solution. Two-dimensional parameter is used, from which projection of processed curve onto represented parametric surface is derived. For specific processed curve boundary conditions are established. This allows to solve elliptical equations in parametric space. In the method of generation of surfaces, based on elliptical partial differential equations boundary conditions are used to develop the form of the surface. This allows to get surfaces in closed form, even for cases of general boundary conditions. In [6] complex approach is suggested, that can include surface generation based on model of potentials. This allows to design flexible topological surfaces interactively and modify generalized boundary conditions, as well as setting various geometrical and physical constraints. This, in turn, allows to maintain different interactive methods and checks on boundaries.

So, problems of verification and problem solving method results analysis for problems of mathematical physics are of current interest, and by today there are no common approach to test and verify problem solving methods for mathematical physics problems with empirical automated research. In this paper developed software solution to automate generation of test models and further verification of convergence analysis, stability analysis and other characteristics of problem solving methods for problems of mathematical physics is discussed.

2. Architecture of software to generate test problems for mathematical physics problems

In order to be flexible about creation of different test problems for various mathematical platforms the architecture of developed software system must have the following properties:

- new problem solving methods for PDEs shall be easy to test.
- new mathematical software packages that use modern math scripting languages shall be easy to add.
- data for further verification and post-processing shall be consistent and well-structured.

To achieve all these, software architecture introduces 3 components: generator of test problems, converter of input data and results verification module. Output results are defined as array of points and corresponding solution value.

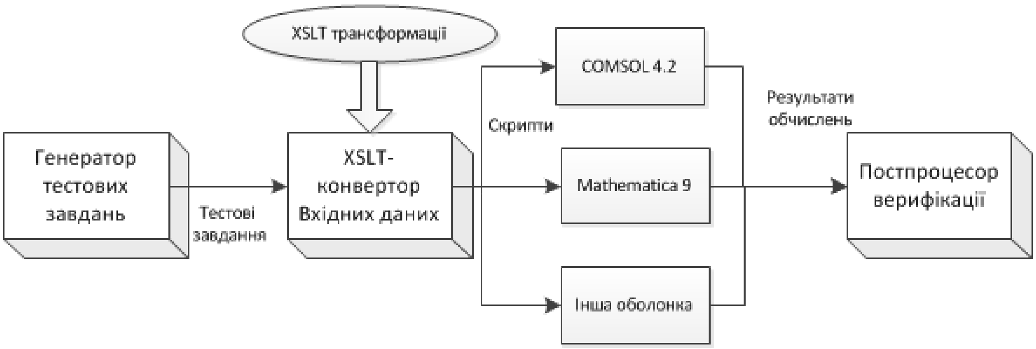

Fig.1. Core software architecture for testing and verification problem solving methods of mathematical physics problems

Input test problems can be defined on client or come from generation system with corresponding constraints. Algorithm of automated generation produces input mathematical models:

- constructs and parameters of initial and boundary conditions;
- PDEs that describe physical processes.

Using XSLT transformations these structures are translated into scripting code of corresponding mathematical software package. To test new problems solving method for PDEs it is required to write corresponding XSLT transformations to get input data for mathematical software packages.

Verification post-processor checks correctness of solution. Also it checks stability and convergence, deviations of produced solutions depending on model parameters.

Definition of test structures includes classes that allow flexible represent various kinds of wide group of problems. Due to large number of elements let's discuss only the ones given on Fig. 2

PDETask class (Fig. 1) is used to define condition of differential equation and contains list of regions and list of initial and boundary conditions. Main requirement for these regions — their continuity and regularity. Boudary class is used to hold boundary conditions, that define equation behavior on the boundary of certain region, that is defined in Domain class and has the following data: interval name, space, type and list of variables. Surface is defined by parametric equations of homeomorphic mapping of one-unit square (or line or point, depending on number of dimensions). Many mathematical software packages give no possibility of such representation, so generator uses surface triangulation algorithm, which is used during translation into mathematical scripts. Relation of composition exists between these entities, as boundary conditions and intervals are integral parts of equation.

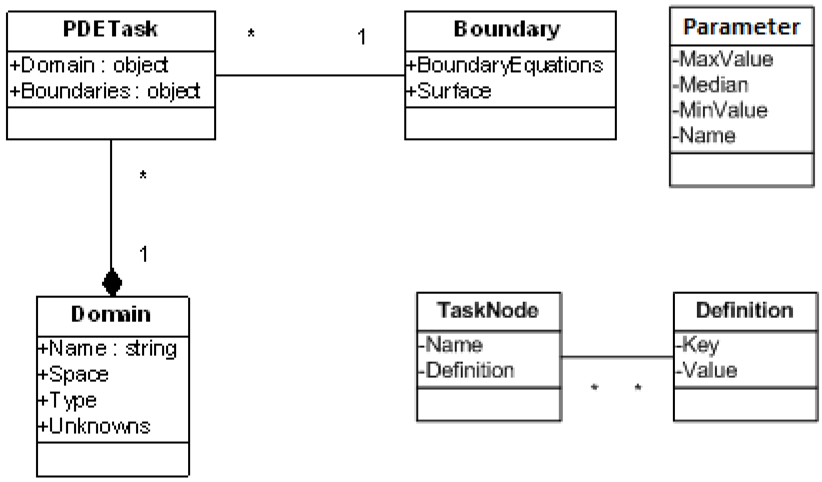

Fig. 2. Core classes of input problem

Let's consider generation of test problems in example of parametric boundary conditions. Boudary conditions are defined by general differential equation on the boundary. Special case are boundary conditions of a third kind, or Robin boundary conditions, which are defined as linear combination of function to solve for and its derivative, which are easy to parametrize to generaqte test problem:

$$\alpha y'(a) + \beta y(a) + \gamma = 0; \qquad (1)$$

For Robin boundary conditions generator changes α and β within predefined intervals, generating different problems. By solving these problems one can research influence of parameters on problem solving method. For general case of boundary conditions parameters that generator can change shall be set manually.

If there's a need to handle surfaces, described by more complex (hyperbolic, parabolic etc.) equations, then certain limitations need to be observed.

We designed software architecture for verification and automated testing of PDEs problems solving methods that observes all these requirements. This architecture allows easy software adaptation to various platforms for mathematical calculations, ensures data are consistent and well-structured for proper operation of software.

3. Results of generation

For algorithm testing we used widespread microelectronics construct with solid connectors. Construct under research consists of aluminum heat-sink (9x140x10мм) and silicon crystal SiO_2 (50x100x10мм), put on heat-sink with 4 aluminum connectors (4x4x10,5мм). Gap between connectors is filled with PVC. Heat sources are established in the center of lower surface of the Chrystal, their sizes are 10x5x1мм, 5x10x1мм, 10x10x1 мм. Input data were tweaked in a way that shows best way functionality of visualization algorithm.

As result of software processing the source code showed below was generated by XSLT transformation. Each expression is added by corresponding XSLT-tag.

Listing. 1. Generated source code for Comsol Script 4.2

```
import com.comsol.model.*
import com.comsol.model.util.*
model = ModelUtil.create('Model');
model.modelPath('D:/Generator/Sample/Simulation 1 - Therma');
model.modelNode.create('mod1');model.geom.create('geom1', 2); model.mesh.create('mesh1', 'geom1');
model.physics.create('emw', 'ThermalMechanical', 'geom1');
model.study.create('std1'); model.study('std1').feature.create('mode', 'ModeAnalysis');
model.geom('geom1').feature.create('r1', 'Block'); model.geom('geom1').run('r1');
```

```
model.geom('geom1').run('r1'); model.geom('geom1').feature.create('r2', 'Block');….
```

Fig. 3. shows 3D visualization of temperature distribution in microelectronic construct by visualization of sections, acquired by solving the problem with COMSOL 4.2.

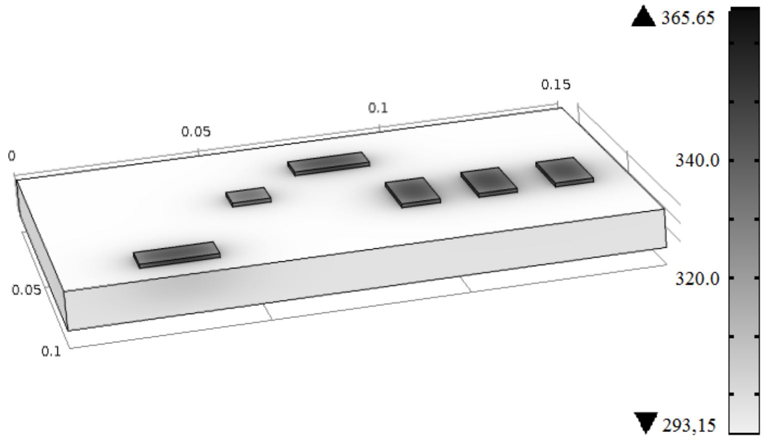

Fig. 3. Temperature distribution visualization on test construct.

Output results are defined as array of points and corresponding solution value.

Conclusion

We have developed interactive preprocessor for generation of test problems for mathematical physics problems with flexible architecture, which allows to verify problem solving methods in the area of mathematical physics. We proposed new approach to verification and automated testing problem solving methods in the area of mathematical physics, which uses existing algorithms of generation of sample input data, modified according to application area. To test new problem solving method, corresponding XSLT transformations need to be defined in order to acquire input data. Input test problems can be set on client level or arrive from system of generation with appropriate constraints. Algorithm of automated generation of test problems for mathematical physics will be considered in upcoming papers.

1. Niermann, Patel HITEC: A test generation package for sequential circuits/ Niermann, Patel// Design Automation. EDAC., Proceedings of the European Conference on.– 1991.– C. 214-218. 2. Cooper, Graham, Sweller, John Effects of schema acquisition and rule automation on mathematical problem-solving transfer / Cooper, Graham; Sweller, John // Journal of Educational Psychology. – 1987.– C. 347-362. 3. Danping Peng, Barry Merriman, Stanley Osher, Hongkai Zhao, Myungjoo Kang. A PDE-Based Fast Local Level Set Method / Danping Peng, Barry Merriman, Stanley Osher, Hongkai Zhao, Myungjoo Kang// Department of Mathematics, University of California at Los Angeles. – 1999.– C. 410–438. 4. Fokas A unified transform method for solving linear and certain nonlinear PDEs/ Department of Mathematics, Imperial College of Science, Technolgy and Medicine, University of London, London SW7 2BZ, UK, Institute of Nonlinear Studies, Clarkson University, Potsdam, NY. – 1997.– C. 1411–1443. 5. Sirendaorejia,Sun Jiong Auxiliary equation method for solving nonlinear partial differential equations/ Department of Mathematics, Inner Mongolia Normal University, Huhhot 010022, Inner Mongolia, PR China. – 2003.– C. 387–396. 6. Haixia Du, Hong Qin Dynamic PDE-based surface design using geometric and physical constraints/ Department of Computer Science, State University of New York at Stony Brook, Stony Brook, NY. – 2005.– C. 43–71.