Software Design of Optimization Laboratory OptiLab

František Mach, Pavel Karban

Faculty of Electrical Engineering, University of West Bohemia, Univerzitní 8, Pilsen, Czech Republic, e-mail: {fmach, karban}@kte.zcu.cz

Abstract **An overview of the software design, methods and data abstraction of the OptiLab framework is presented. The framework is aimed at the advanced analysis and optimization of mathematical models of physical fields and also coupled problems. The algorithms of the framework are implemented with respect to solution demands and huge data requirements. The paper presents a short description of the framework, its basic concept and structure.**

Keywords **software design, data abstraction, framework, optimization, sensitivity analysis, uncertainty analysis**

I. Introduction

OptiLab is a new framework for advanced analysis and optimization of mathematical models described by partial differential equations and characterized by a strong computational complexity. The framework is written in C++ and Python and works closely with Agros2D [1]. The development of OptiLab framework was initiated mainly by the need for a software tool for research of novel analyses and optimization methods in coupled problems.

The framework is not designed to solve certain applications. It is intended to serve as a generally purpose tool containing whatever is needed for analysis and optimization of mathematical models. The main features of the framework are:

- parameter study of input parameters,
- sensitivity and uncertainty analysis,
- single-objective and multi-objective optimization.

The paper presents an overview of the main concept and structure of the OptiLab framework, data abstraction and features.

II. Basic Concept and Structure of the Framework

OptiLab consists of an object-oriented library *variant* and graphical application for management of solved studies and powerful analysis of results. Library *variant* is intended for efficient administration of a large set of computer models, their numerical solution on a local computer or computer cluster and also contains all algorithms for model analysis, optimization and also operations over results. The basic structure of the library *variant* is show in Fig. 1 by a simplified dependence graph.

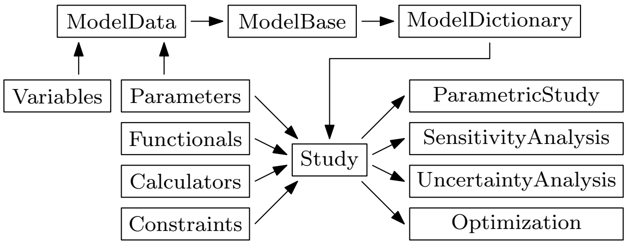

Fig. 1. Dependence graph for main classes of *variant* library

The class *ModelBase* is intended for the description of the solved models. It separates the computer model from other parts of the system and defines it by input parameters, output variables and also mathematical model. For an efficient administration of the set of computer models in the form of instances of the class *ModelBase*, a family of *ModelDictionary* classes are proposed. Any object of this class allows performing operations over the whole set of models, i.e. their solving, their saving, their repetitive reading, filtering with respect to specified criteria, extracting parameters, variables across all the models and also logging all information about models and study.

The class *Study* defines the structure of objects necessary for both the analyses of computer models and optimizations. Main role of *Study* class is to implement common features for all kinds of studies (operations with functionals, analyzed parameters, calculators, criterions, etc.). All particular classes such as *ParameterStudy*, *SensitivityAnalysis*, *UncertaintyAnalysis* and *Optimization* are inherited from class *Study* and supplement it with the algorithm of the analysis itself.

III. Conclusion

The paper presents a short overview of the basic concept and structure of the OptiLab framework. The proposed object-oriented concept of the *variant* library allows connecting different kinds of analyses or optimizations and performs both tasks on the same set of the models.

At present, the methods of the sensitivity analysis based on OAT (One-At-Time), Morris methods, correlation methods and also the Sobol approach are implemented in the system. As for the optimization algorithms, implemented are the conjugate gradient method, genetic algorithms (NSGA-II and also its own modification) and simulated annealing.

IV. Acknowledgements

This work was supported by the European Regional Development Fund and Ministry of Education, Youth and Sports of the Czech Republic (project No. CZ.1.05/2.1.00/03.0094: Regional Innovation Centre for Electrical Engineering - RICE), by the Czech Science Foundation (project GACR P102/11/0498) and project SGS-2015-035 at the University of West Bohemia.

References

[1] Karban, P., Mach, F., Kůs, P., Pánek, D., Doležel, I.: ”Numerical solution of coupled problems using code Agros2D”, Computing, Vol. 95(1), pp. 381-408. 2013.