SoftwareX 6 (2017) 231–236

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original Software Publication

Optimel: Software for selecting the optimal method

Olga Popova [a],[*], Boris Popov [b], Dmitry Romanov [a], Marina Evseeva [c]

[a] Kuban State Technological University, Institute of Computer Systems and Information Security, 2, street Moscow, Krasnodar, Russia
[b] Kuban State Technological University, Institute of Oil, Gas and Energy, 2, street Moscow, Krasnodar, Russia
[c] Kuban State University, Department of Romano-Germanic Philology, 149, street Stavropolskaya, Krasnodar, Russia

ARTICLE INFO

Article history:
Received 20 February 2017
Received in revised form 1 August 2017
Accepted 2 August 2017

Keywords:
Optimization methods
Domain
Knowledge structuring
Exploratory study

ABSTRACT

Optimel: software for selecting the optimal method automates the process of selecting a solution method from the optimization methods domain. Optimel features practical novelty. It saves time and money when conducting exploratory studies if its objective is to select the most appropriate method for solving an optimization problem. Optimel features theoretical novelty because for obtaining the domain a new method of knowledge structuring was used. In the Optimel domain, extended quantity of methods and their properties are used, which allows identifying the level of scientific studies, enhancing the user's expertise level, expand the prospects the user faces and opening up new research objectives. Optimel can be used both in scientific research institutes and in educational institutions.

Code metadata

Current code version	V1.0.1
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-17-00017
Legal Code License	GNU General Public License, GPL/LGPL
Code versioning system used	git
Software code languages, tools, and services used	Lazarus IDE v1.6, Free Pascal
Compilation requirements, operating environments & dependencies	Windows, Linux
If available Link to developer documentation/manual	https://github.com/popova-ob-77/Optimel/wiki
Support email for questions	popova_ob@mail.ru

1. Motivation and significance

The objective of creating the Optimel software is to automate the process of selecting the most appropriate optimization method of the set of all known optimization methods. In order to obtain the software, the following tasks were solved. The method for representing the "optimization methods" domain was developed and initial data including all known optimization methods were collected. The "optimization methods" domain was compiled, tested and verified as a binary tree of question-answer system.

The Optimel software is important because it saves time and resources by conducting an exploratory study to select the solution method for an optimization problem. Before any study, Text mining has to be conducted, the information sources have to be read and the obtained information has to be structured using the intellect. Next, with the knowledge available, the required solution method for an optimization problem has to be selected. The efficiency of several methods may be compared as well. So, the time to be spent for the exploratory study in order to select a solution method for the current optimization problem equals:

$$t = t_1 + t_2 + t_3 + t_4,$$

where t_1 is the time of searching for primary and secondary sources;

t_2 is the time to analyze the selected literature;

t_3 is the time spent for the process of understanding the material and selecting several methods for comparison;

t_4 is the time spent for comparing several solution methods for picking the most efficient method.

The Optimel software reduces the time t considerably by automating the selection of optimization problem solution method at the expense of the following tasks completed in advance: literature review conducted in the "optimization methods" domain; the process of selecting the most appropriate option of all known ones being included in the question-answer system tree developed. Hence

[*] Corresponding author.
E-mail addresses: popova_ob@mail.ru (O. Popova), khizova2004@mail.ru (M. Evseeva).

http://dx.doi.org/10.1016/j.softx.2017.08.001
2352-7110/© 2017 Published by Elsevier B.V.

the Optimel software yields a smaller total time $t_1 + t_2 + t_3 + t_4$ than the t which currently exists. In it, the pre-structured material for the domain is employed. The user of the software does not have to compare the methods as while traversing the tree the most relevant solution method of the set of all known ones is identified.

For Optimel, the source data are the properties of the optimization problem for which a solution method has to be found. Unlike the known search engines that give information according to the user's query, it is up to Optimel to ask questions to the user—ones that are worded in a special way. By answering the questions put by the software, one can render an obscurely worded problem more precisely and in the end obtain a solution method for it. The user will get a justification of exactly this problem solution method that is the most appropriate for the problem.

1.1. Practical novelty

At present, the time t_1 and t_4 can be significantly reduced using the state-of-the-art technical solutions available, e.g. Text Mining, Web Content Mining [1–3], Scopus, Google, Wolfram|Alpha [4], expert systems [5,6] (CLIPS, OpenCyc), applied software packages for optimization calculations (Maple(software), MATLAB, Mathematica, XPRESS) [7,8]. The applied software packages are efficient if the user knows exactly what optimization method fits best for solving the problem.

The suggested Optimel software unites all these stages using the initial data on the domain where the required optimization methods is looked for collected in advance. After this, one of the applied software packages mentioned above can be used for optimization calculation.

1.2. Theoretical novelty

In the Optimel software, a new logical model of domain description is suggested which differs from the models presently known as follows: $M = \langle T, P, M_x \rangle$, here T is the set of descriptions of various problem solution methods and P is the three rules [9] allowing one to build structured knowledge about a domain of set T, e.g. "optimization methods", that are close in their shape to the human intellect knowledge representation structures; and M_x is the mathematical object [10].

In Optimel, the formal language is determined by the principle of human intellect information structuring and by mathematical object M_x: a set of structured domains, each of which can be represented by a binary question-answer system tree and is taken from the human intellect where they are stored as structured knowledge and are a well learned knowledge.

The appearance of the binary question-answer system tree for the "optimization methods" domain and examples of selecting the most appropriate problem solution method are presented in the video clip and binary tree of Q&A system.xlsx file attached to the paper.

2. Software description

The software has an open and clear code, it is quick to launch and it features a clear user interface: all one needs is to read the question text and answer it by choosing from the "yes" or "no" options. While working on important stages, some buttons become invisible while other button sections appear (Figs. 1–3). This helps facilitate working with the software.

The software implementation of the binary question-answer system tree is clear and simple, which allows for programmer easily modifying the software code as new optimization methods appear. Meanwhile, the tree structure is not reformed but new elements are added into it, with its integrity maintained.

Likewise, the user can browse the entire structure of the binary question-answer system tree expressed in the second method keys. Here questions are replaced by keys because so the entire "optimization methods" domain can be seen at once, structural links can be checked, the domain can be tested and verification can be conducted. All this allows ensuring a valid result to the user. The user can also check the solution obtained using the tree and Optimel wiki. The tree is located to the left of the results output window and it can be navigated from root to leaf in any direction.

2.1. Software architecture

The software architecture is described below, with each block discussed in more detail. The software consists of the following blocks: Block 1—knowledge base block; Block 2—the Block of DataBase Dictionary; Block 3—the block of user dialogue work section. The interaction of the blocks can be seen in the timing diagram of the software product (Fig. 4).

2.1.1. Knowledge base block

First, the tree is loaded. To do this, the user has to press the button "Load binary tree of question-answer system" (Fig. 1). The standard procedure of elements adding into the binary question-answer system tree is used; the tree being implemented by means of pointers, it features simple implementation and a clear code, and the questions and optimization methods from block 3 are added to the tree in a sequence. The tree does not need to be reloaded if one has to use Optimel again—all one needs to do is go to the tree root. For this, the button "Back to the first question" is pressed which puts the pointer to the tree root and outputs the first question in the solution field (Fig. 2).

2.1.2. User dialogue work section block

The user can now answer the questions using the "Answer choice option" section (see Fig. 3). Questions corresponding to the traversed tree node are displayed in the data output window one after another, depending on the "yes" or "no" answer selected by the user. The current selection determines the direction of tree traversing and the next question. The result will be the optimization method. Examples of tree traversing can be seen in the video clip attached to the paper. Once tree traversing is completed, the "Answer choice option" becomes invisible and the "Back to the first question" button appears.

2.1.3. The block of database dictionary

This block consists of two sections, "Database questions" and "Optimization methods". In the first section, there are all questions characterizing the "optimization methods" domain. For obtaining the questions, the authors used the developed by them theory of obtaining the root and intermediate nodes of the binary question-answer system tree [9,10].

The sketch of the "optimization methods" domain was drawn in paper. Each tree node was marked as a rectangle containing the properties worded as a question. Each rectangle was connected by its structural links with other rectangles. As a result, the complete domain knowledge map as a binary question-answer system tree was obtained which is also the decision-making tree. The sketch amounted to 20 handwritten text sheets. Next, three methods of tree keys placement were applied. The first method involved placing the keys in the direct order from 1 to 127 in the rectangles in the sketch sheets. With the second method, the keys were distributed in the tree nodes at the increment of 20 so as to obtain a sorted sequence with the tree traversed symmetrically. Under the third method, first of all, the intermediate tree node indices were placed beginning from one at the increment of 1 and using the algorithm: