Educational system support to decision-making theory

Oleksii Voloshyn
Taras Shevchenko University
Kyiv, Ukraine
e-mail: olvoloshyn@ukr.net

Daniil Kovaliov
Taras Shevchenko University
Kyiv, Ukraine
e-mail: daniil.kovaliov@gmail.com

ABSTRACT
The issue of the software support to the decision-making area training courses in the form of an educational and evaluating system is considered. Here we outline the system's purposes, its design, structure and functionalities. The article pays attention to the system internals, such as the rules of fuzzy output during the process of complex task of knowledge estimation of the user. The perspective future work is also discussed.

Keywords
Information technology in education, decision-making theory, evaluation, team development.

1. INTRODUCTION
We consider the software system SMDM (СМПР) that supports the educational course «Systems and Methods for Decision Making» lectured to the third-year students studying Informatics at the Cybernetics Faculty of the Taras Shevchenko National University of Kyiv (Ukraine). The system is built according to the traditional course tutorials [1,2]. Due to space limitation here, we will concentrate our attention to the SMDM solely, not entering into the broad area of e-learning models and systems.

The software development started in 2005-2006, aiming to bring the course to an e-learning environment. The first version, SMDM-06 system was represented at the MeL-2006 (Modern (e-)Learning) conference (Varna, 2006) and published in [3]. This early version is composed only of a bunch of separate program modules. These modules bring the realization of different algorithms providing solutions of definite decision-making problems according to theory [1,2]. Program modules (developed by students during the laboratory studies) were worked out independently and created using different programming languages, having a random structure and interfaces.

The systems integration into an interacting system of modules was implemented in version SMDM-08 by forming unified requirements to program realization using C# and .NET platform in MS Visual Studio environment with collective development technology SVN [4]. During 2005-2012 more than 60 students in a form of an open project were involved in SMDM developments. The SMPR- 08 version system can be called as «educationally-methodical and demonstrationally-testing». While demonstrating the solution of a definite DMT problem user could check his own solution by inputting the answer and getting a «Correct» or «Incorrect» mark. In SMDM-10 (represented at MeL-2010 (Kiev, 2010) and published in [5]) the evaluation function for students' knowledge estimation was implemented. This version includes an extended interface, specifically, from the point of view of the languages used - Ukrainian, Russian, English and Chinese (separate modules). SMDM-10 as a supplement to the tutorial [3] is already used in a number of Ukrainian universities (for example, [6]).

Version SMDM-12 represented in this article can be considered as an «industrial» version of the system. Comparing to SMDM-10 in SMDM-12 two main functionalities were incorporated: 1) evaluation of students' knowledge by the whole DMT course, set in «clear» or «fuzzy» forms [5]; 2) implementation of a software protection system, designed to prevent unauthorized usage.

In the next SMDM version the authors plan to bring the system to an online environment, transferring it to the Internet. Also publication of tutorial in English and Russian is arranged (Chinese is in perspective). Ideally, SMDM system should become the basic part of Educational Internet project, developed at Taras Shevchenko National University of Kyiv like the famous coursera.org and udacity.com do.

2. GENERAL INFORMATION ABOUT SMDM SOFTWARE SYSTEM
System consists of a core, and a set of specialized modules responsible for solving definite set of subject area problems. Core system creates an environment for module functioning with the ability to solve problems in a parallel manner, and with data exchange between modules. The core represents the general interfaces, data exchange standards; help subsystem functionality and the modular

The core part is made over a common buffer — an instrument of data exchange between modules. Buffer can manipulate with source data of different kind, and can contain the result of the data procession. It represents «environment» for system level data, data that can be accessed from any module. On architectural level buffer is a special class, that provides interior modules with interface for upload, store and validation of data.

Core doesn't specialize for a concrete algorithm of concrete educational course. Specification of tasks is provided through the external modules, which are available to core to use. Precisely this property of the system core allows to use it as the basis for the development of training systems for a variety of training courses with the structure "classes of problems - the methods of their solutions" with the ability to exchange data between classes.
The structure of the module responsible for a particular class of problems is constructed in such a way that the kernel can auto-detect not only the presence of the module, but also - some of its characteristics (period of use, the number of uses, etc). Since the system consists of a core and a plurality of modules provided in the form of plug-ins, the presence or absence of a single module does not affect the operation of other modules of the system.
"Head" of the program includes bookmarks: 1) "File", which allows you to quickly close the program; 2) "Modules", which helps to run an appropriate module; 3) the "language" that allows you to switch the interface to one of the four languages (Ukrainian, Russian, English, Chinese); 4) "Help," which has two divisions: 4.1) "About", which brings up a window with full information about the developers of the project , and 4.2) "modules", which opens

up an additional menu. Help is assigned to each of the connected modules, and provides information on the use of the module with its theoretical material.

SMDM presents each topic under discussion as a separate module, - in a separate window. Data can be entered from the keyboard, and from the buffer alternatively. Each module has a system content designed to teach and test the knowledge of a student. Currently the system has 8 sections on DMT (in accordance with [5]: 1. Conflicts and compromises; 2. Multi-criteria optimization; 3. Cooperative decision-making (DM/ПР); 4. Voting methods; 5. Expert evaluation; 6. DM in the face of uncertainty; 7. DM in fuzziness; 8. Collective utility function) and one module of "psychological tests" [7].

The authors hope that colleagues (teachers and students) from other universities (primarily Ukraine, Armenia and Russia) may be consolidated in using the software system SMDM in the learning process. The systems tools (core) and improvements (handling, processing, substitution) already existing in the system may be used to incorporate new algorithms in subject areas.

3. KNOWLEDGE MANAGEMENT SYSTEM

Each module implemented in SMDM corresponds to one particular DM approach and it offers the student with a particular problem postulation, giving a task, for example, - finding optimal, according to some criteria strategies, identifying the required set of states of the system, etc. Tasks are generated automatically, to avoid repetition, and may have varying complexity (in terms of the problem dimensionality, of number of states, or members, etc.).

All information on evaluations obtained during the passage of tests is stored in the system and can be obtained by the user at any time in the form of a window with the general results of testing on all sections of the course, together with recommendations visualized on corner areas indicating the topics to be better learned. One more window is appearing with the general results of the course and with recommendations on the materials to be studied better.

Because of the different modules comprising the testing differ significantly from each other in their type and complexity, a specific mathematical model is used in systems design, which takes into account the parameters of tasks under consideration. Along with the number of correct responses the task complexity and its execution time are also considered as parameters. The complexity level of task defines the limits of the allowed duration for a task and the maximum number of the allowed errors.

If the task was completed faster than the time required to perform, the student receives a higher rating, with increase of time, and the student begins to lose time points when he does not get task completed at a time. Significance was set at 1, but can be increased (student motivation to perform tasks faster for more points). Values are defined by an expert estimate (including statistics on the results of testing) for each module of the system taking into account the complexity of the

The level of difficulty of task is given by the linguistic variable CM = {very low (VL), low (L), medium (M), high (H), very high (VH)}. The "correct answers" are given by the linguistic variable NCA = {very low (VL), low (L), medium (M), high (H), very high (CH)}. For example, the above values of linguistic variables can be set between 0% and 50% to 10% discrete, in the range of membership functions, linear with values in [0,1]. Let each "estimates interval level of confidence" correspond to a certain score, such as "4-point scale": "unsatisfactory", "satisfactory", "good", "excellent" (as a rule, these estimates correspond to the following values in the 100 - point scale: [0.59], [60.74], [75.89], [90,100]).

User of SMDM system offered a "standard" base inference rules evaluation of testing, containing, for example, the rules of the type - "if CM = H and NCA = L, then score =" unsatisfactory" with the appropriate "level of confidence" (determined by a given algorithm, fuzzy logic, which can be selected by the user from the corresponding base [8]). The result of the algorithm is a real number in the interval [0,1] or an integer in the range [0,100] - "the exact degree of certainty" assessment of student knowledge.

Optionally, the user based on the received accurate ("clear") mark can get "fuzzy" mark, selecting from a database of fuzzy inference rules certain rules (or by using the "standard" basis) of the form: "if clear mark is in the range [60,65], your score = "solid three". "

4. CONCLUSION

The purpose of the report – is in "advertising" the instructional testing module that evaluates the students knowledge by software system SMDM, developed at the Department of Cybernetics, Kyiv National Taras Shevchenko University. The system is used by third year students of several generations studying "Computer Science" under the guidance of the authors of this article. SMDM is a complete software package to support the training of courses on the theory of decision-making, the laboratory work, testing of students and providing objective (automatic) assessment of their knowledge on tests and exams. Since the "core" of the system SMDM is not specialized for the specific algorithm and/or a specific training course, it can be used as a tool for the development of software products for more training with the structure "classes of problems - the methods of their solution" with the ability to exchange data between classes.

REFERENCES

[1] A. Voloshin, S. Mashchenko, “The theory of decision making. Textbook”, *Kiev: Publishing center "Kiev University",* 304p. 2006 (in Ukrainian).

[2] A. Voloshin, S. Mashchenko, “Models and methods of decision-making. Textbook”, *Kiev: Publishing center "Kiev University",* 336p., 2010 (in Ukrainian).

[3] O.Voloshyn, K.Berezovskiy, I.Drozdov, “Developing collective teaching computer software for the course Decision theory”, *International Journal "Information Technologies & Knowledge, Vol.1, N1,* pp.33-36, 2007.

[4] E.Gamma, R.Helm, J.Vlissides, “Elements of Reusable Object-Oriented Design. Design patterns”, *Moscow: Publishing House "Peter",* 2006 (in Russian).

[5] A. Voloshin, D.Kovalev, “Educational and methodical testiruyusche-evaluator software system to support courses in decision theory”, *Problems of Computer Intellectualization, ITHEA*, pp.293-298, 2012 (in Russian).

[6] G.Gaivoronskaya, A.Bondarenko, “The task of choosing the topological structure of the network access”, *Refrigeration and technology, № 2*, 2013 (in Russian).

[7] A. Karelin, “Great Encyclopedia of psychological tests”, *Moscow: Penguin Books,* 416 p., 2007 (in Russian).

[8] V. Snytyuk, “Forecasting. Models, methods and algorithms”, *Kiev, Maklaut,* 364p., 2008 (in Ukrainian).