Web-based Analytical Decision Support System

Martti Sutinen[1,2] – corresponding author
[1]*Dept. of Engineering Physics and Mathematics, Aalto University School of Science and Technology,*
02150 Espoo, FINLAND
martti.sutinen@tkk.fi
+358400634208

Mats Danielson[2], Love Ekenberg[2,3],
and Aron Larsson[2,3]
[2]*Dept. of Computer and Systems Sciences, Stockholm University / KTH, Forum 100, SE-164 40 Kista, SWEDEN*
[3]*Dept. of Information Technology and Media, Mid Sweden University, SE-851 70 Sundsvall, SWEDEN*
{masu, mad, lovek, aron}@dsv.su.se

***Abstract*— This paper presents a web-application supporting structured decision modelling and analysis. The application allows for decision modelling with respect to different preferences and views, allowing for numerically imprecise and vague background probabilities, values, and criteria weights, which further can be adjusted in an interactive fashion when considering calculated decision outcomes. The web-application is based on a decision tool that has been used in a large number of different domains over the last 15 years, ranging from investment decision analysis for companies to public decision support for local governments.**

Keywords: Decision support systems; Web application; Web-based analysis.

I. INTRODUCTION

During recent years of rather intense research activities several decision analytical methods have emerged as alternatives to classical utility theory. One reason has been that the latter is felt to require too much of the decision maker and therefore to be unrealistic. In particular, first-order approaches, i.e., based on sets of probability measures, upper and lower probabilities, or interval probabilities, have prevailed. A main class of such models has been focused on expressing probabilities in terms of intervals. Since the beginning of the 1960s the use of first-order (interval-valued) probability functions, by means of classes of probability measures and upper and lower probabilities, has been integrated into classical probability theory by a variety of researchers. Similarly, upper and lower previsions have also been investigated by various authors. A few approaches have also been based on logic.

A common characteristic of the first-order representations above is that they typically do not include all of the strong axioms of probability theory and thus they do not require an agent to model and evaluate a decision situation using precise probability and value estimates. An advantage of representations using upper and lower probabilities is that they do not require taking probability distributions into consideration. On the other hand, it is then often difficult to devise a reasonable decision rule that finds an admissible alternative out of a set of alternatives and at the same time fully reflects the intensions of a decision maker. Since the probabilities and values are represented by intervals, the expected value range of an alternative will also be an interval. In effect, such a procedure retains all alternatives having over-lapping expected utility intervals, even if the overlap is very small. Furthermore, they do not admit for discrimination between different beliefs in different values within the intervals. Even more problematical, these theories tend to be considerably more complicated than classical utility theory and has therefore become of quite limited use in practical decision making.

In any case, all of these representations face the same trade-off. Zero-order approaches (i.e. fixed numbers representing probability and utility assessments) require unreasonable precision in the representation of input data. Even though the evaluation and discrimination between alternatives becomes simple, the results are often not good representatives of the problem and sensitivity analyses are difficult to carry out for more than a few parameters at a time. First-order approaches (e.g. inter-vals) offer a remedy to the representation problem by allowing imprecision in the representation of probability and utility assessments, reflecting the uncertainty inherent in most real-life decision problems faced by decision makers. The trade-off between realistic representation and discriminative power has not, though, been solved within the above paradigms. For a solution, one could look at second-order approaches allowing both imprecision in representation and power of admissible discrimination.

Approaches for extending the interval representation using distributions over classes of probability and value measures have been developed into various hierarchical models, such as second-order probability theory. In general, very few have addressed the problems of computational complexity when creating tools for solving decision problems involving such estimates. Needless to say, it is necessary to be able to determine, in a reasonably short computation time, how various evaluative principles rank the given options in a decision situation.

We have over the years been working with this and developed a framework for decision making considering all these aspects. We have incorporated possibilities to express vagueness and hierarchical orders regarding probabilities, utilities, and criteria weights. We have also investigated how rational decision processes can be formulated and how expert advices can be modelled even in political sensitive decision making. Further, we have considered the aspects of computational complexity and consequently developed algorithms for solving multi-linear problems containing thousands of consequences in a very short time, thus enabling the construction of an interactive decision analytic support tool. The most concise description of the latter is provided in [8].

II. THE TOOL

Our work has been implemented in DecideIT, which supports decision making in complex problems and environments. Danielson et al. [6] provides a short introduction to an earlier version of this tool. Some extended versions can be found in [9, 10]. Its development is the result of long-time research at the Department of Computer and Systems Sciences (DSV), Stockholm University and the Department of Information Technology and Media, Mid Sweden University (ITM). Prior to 2002, it was mostly used for research purposes, when the company Preference [26] was founded to promote the product and accompanied services commercially. DecideIT is currently offered for businesses with a yearly license fee and consultation expenses and is available for free for academic purposes. The software can be used in order to evaluate decision problems where risks, uncertainties, and conflicting objectives exist. Consultation is often requested for help in problem structuring and interpreting the output.

DecideIT can be viewed as the implementation of the so called DELTA method for computational decision analysis with imprecise information, see e.g. [7, 19]. The DELTA framework is based on classic decision analysis, i.e. the application of theories of rational choice on decision problems within (real-life) contexts, see, e.g., [3, 15]. Central in decision analysis is the utilization of normative decision rules such the maximization of expected utility and various means for sensitivity analysis and the handling of uncertainty and conflicting objectives (multiple criteria). Applications of decision analysis are common within policy analysis and large-scale decision problems within authorities and corporations. The DecideIT decision tool has been used in a large number of different domains ranging from investment decision analysis to public decision support and even procedures for efficient demining, described in e.g., [11, 20].

The DecideIT software code consists of a native library written in C/C++, which covers most of the algorithms of the DELTA method used for analysis, and an interface written in Java. These elements operate together through the use of the Java Native Interface (JNI). C is a machine dependent programming language whereas Java works independent of the environment. This means that there needs to be a native library for each operating system / computer architecture combination and a runtime environment (JRE) to handle the Java code. This kind of combined application is often converted to native applications (such as .exe files on MS Windows) with the help of tools such as, e.g., Excelsior JET [12].

III. WEB-BASED DECISION SUPPORT

Web-based decision support systems (WB-DSS) differ from the traditional decision support systems (DSS) in that they are globally accessible on the web and that they utilize databases and models which can be applied to various user groups. Often, WB-DSSs are implemented as either simplified versions of desktop DSSs or as specialized targeting systems used by e-commerce to boost sales. WB-DSSs have limited interfaces to ensure fast response times and interaction is kept at a minimum to enable users to focus on their main tasks. Privacy and security concerns need to be considered when handling personal information. [2]

The biggest challenges in the development of WB-DSSs are limitations in bandwidth and interactivity, need of training in solving complex problems, the diverse target audience, and trust, security, and privacy issues [2]. It can be argued that some of these problems can be remedied by dividing data management and calculations meaningfully between the server and the client. Our approach, as discussed below, enable the users to manage their own data, removing bandwidth, trust, security and privacy issues. Also, the calculations are made locally so there is no loss of functionality compared to the desktop DSS. There will still be a need for training, but that is inherent in complex problems.

DecideIT cannot be considered a typical WB-DSS as it is a comprehensive analytical tool that requires strong participation by the user. While specialized WB-DSSs used in e-commerce are suitable for automatically identifying the user’s preferences and promoting relevant products, they cannot be applied to structuring decision problems and presenting analysis on values and risks. In our case, emphasis is placed on computational and analytical ability, interaction and security.

Some analytical web tool implementations exist, notably JavaAHP and Web-HIPRE. JavaAHP is based on the analytical hierarchy process (AHP) [27] and enables the user to structure a problem with several objectives organized in hierarchical levels that are compared pair-wise to find the importance weights. Alternatives are similarly compared pair-wise with respect to each lowest level criterion. JavaAHP is suitable for limited multi-criteria problems in which users are able to compare objectives and alternatives in a precise manner. The number of required pair-wise comparison rises significantly as the number of criteria and alternatives increases. [31]

The other analytical tool, Web-HIPRE, is also based on AHP. The same issues apply to it as well, mostly the complexity in problems with numerous criteria and alternatives and the lack of modelling uncertainty. However, Opinions-Online [13] can be used in group decision making to evaluate criteria and alternatives. The differences in opinions can be taken into account by weighting different stakeholders' assessments. While representing the decision situation well, the intervals of the different views are not reflected in the final scores and must be tested with sensitivity analysis. [22]

DecideIT aims at remedying the lack of uncertainty representation in multi-criteria problems as well as decision trees. It allows the users to provide interval, likeliest point, and relation statements about weights, probabilities, and consequences. Decision trees can be connected to criteria supporting effects of random events even in multi-criteria models. The alternatives can then be analyzed by examining value intervals on a certain contraction level and sensitivity analyses show the effect of uncertainty in weights and probabilities on the alternatives' values. [19]

There are various ways in which web-based decision analysis tools can be supported. A decision making process could start with problem or value identification, depending on the view [16]. In either case, after the problem has been defined, alternatives should be identified. One way to develop alternatives and determine their value in decisions concerning multiple stakeholders is to use the RPM Screening Process in which ideas are generated, commented, evaluated and analyzed iteratively to produce an optimal portfolio [18]. The RPM Screening Process does not have to be used in its entirety; it can be used merely as a pre-stage process to remove the dominated alternatives and to evaluate alternatives in group decision making situations. A similar collaborative approach might be taken to determine criteria and value relations in addition to probabilities of the different random events to accompany expert reviews. DecideIT can then be used to build a model that allows for the different opinions and represents uncertainty in analysis. The effect of uncertain information or difference of opinions can be seen in sensitivity analyses. It is important to notice that additional information and negotiations between stakeholder groups often cost and one should find out whether an uncertain element offers enough value for the decision to be specified [3].

IV. APPROACH FOR DISTRIBUTION

Although the use of the Java programming language requires an extra effort in compilation, the choice of using Java is based upon the following rationales. First, Java is faster to work with from the developer point of view and has extensive libraries. It is a bit heavier for the computer to handle than C, but all the essential algorithms are optimized in the native C library. Also, the need to use expensive compilation tools can be avoided. Java needs the Java Runtime Environment (JRE) installed on the machine. JRE is not always installed and its use is not as straightforward as the use of stand-alone applications.

A method needed to be found to make the distribution of the software easy and the (possible) installation process of Java automatic. Five options were identified:

- Using Java Web Start to download and run the application
- Using Java Plug-in for browser based use
- Using AjaxSwing for browser based use
- Using HTML/XML/JS for browser based use
- Distributing resources in a compressed file

Out of these, only two were good candidates. Distributing a compressed file would be too laborious for the user who would need to know how to use Java commands to run the application. An HTML/XML/JS implementation of the interface would limit it too much and would take considerable time to develop [14]. AjaxSwing [5], previously called WebCream, generates HMTL/JS at runtime automatically, but is not very responsive with advanced interfaces and costs a lot [4].

Then Java Plug-in and Java Web Start (JWS) remains as alternatives. Java Plug-in [23] requires an applet that runs on the browser. Most browsers support Java Plug-in, installing it automatically when needed for the first time. Java Plug-in is very convenient for the user, but it doesn't work offline and requires reprogramming of the interface to an applet [4]. JWS [24] is a technology that enables graphical Java applications to be easily distributed over the Internet. It works by giving the user a JNLP file which points to the location of the needed resources that are subsequently downloaded. The pointer file can be used to launch the application even offline and once the user is back online, updated files are downloaded automatically [28]. For those reasons, JWS was selected, as only minor modifications were needed.

JWS offers some significant advantages to a WB-DSS over the use of browser based technologies. Downloading the application means that all files and calculations can be executed locally. First of all, this makes the user more confident, knowing that the data is not exposed to other parties and that the data is also not limited by Internet connection bandwidth. Secondly, the interface can be much more expansive, interactive and reactive, not limited by the browser's capabilities or the computing power of the server side computer. This approach does, however, affect the mobility of the system, an issue discussed in the conclusion. [2]

V. RESULT

First, JWS was tested with Java Native Interface (JNI) in a simple program. The test program was made on Netbeans [25] Integrated Development Environment (IDE) with the help of MinGW and MSYS [21] for compilation of the C code. As the program worked, it was extended to use basic input and output (IO) and a graphical user interface (GUI) as well as external resources. JaNeLA [30] was used to check the pointer file syntax.

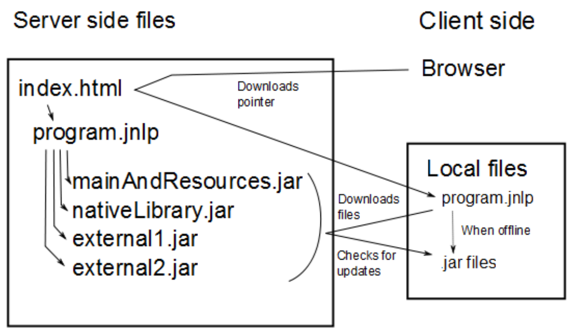

Figure 1. The Java Web Start Principle

Ensuing that, some modifications were made to the original Java source code. These were mostly related to commands pointing to resources. The JAR files, in which all the resources lie, had to be signed in order to enable IO [1]. Also the JNLP MIME type had to be defined for the server [28]. A JavaScript code from Sun was used to identify and automate the JRE installation [29]. The JWS principle is shown in Fig. 1.

VI. USE CASES

The users can be divided into roughly two categories with respect to requirements: (i) consultants and businesses, and (ii) researchers and developers. Businesses need to make fast decisions with uncertain information. Common to these users is that they want fast responses, good estimates and reliable consultation along with security. Researchers often have a well-defined problem and lots of data, but want to know how the methods work, that is, they want documentation on underlying theories.

A typical business usage could start off in various conferences and business events. The software might be first used only by consultants to analyze a smaller decision. Once the customer is convinced about using decision analysis to support decision making, bigger problems are identified, structured and represented with the software. At this point, the customer may want to use the software also independently and buy a license. The program can then be easily downloaded with JWS and can be run from anywhere, provided there is a JRE installation or the means to install it.

A typical researcher will appreciate the free license and use the program to build decision analysis cases or model decision problems at hand. Documentation is then important to assess method possibilities in detail. The program includes a help menu which gives a fast overview of the approach, and there are multiple papers written on the theory behind the tool that can be easily found on the Internet. Creating a forum where ideas and problems are exchanged might help to promote academic use and to find errors and development issues in the software.

VII. CONCLUDING REMARKS AND FUTURE WORK

In this paper, we have discussed the relevance of WB-DSSs in decision making. We have reviewed the opportunities and challenges in developing a WB-DSS and discussed some existing analytical tools. We have identified the need for an analytical WB-DSS which can represent the uncertainty in real-life situations and discussed different approaches for transforming a desktop DSS into a WB-DSS. We have chosen to use JWS as the technology for web-based distribution. JWS allows us to preserve the interactive interface from the stand-alone tool and the fast calculations needed in complex decision situations. However, mobility can and should still be enhanced.

Mobility of the system can be greatly increased by allowing the users to store files on a server even if working with them locally. This can be implemented with a user database that the application connects to. Logging on as a user will also remove the need to keep a license key on every new computer. Users can even establish networks through the server in which information is shared. A simple browser interface could be developed for DecideIT to be used for quick problem solving in conjunction with other tools such as the ones discussed in the paper and as an alternative for downloading the application. This could be a start for building a complete web-based analytical decision support system.

Finally, the computational kernel can be offered as a service for other businesses. This requires a server with support for high-speed connections to ensure fast data transfer and enough computational power for the calculations. For example, e-commerce start-up companies could be interested in using existing systems.

Unfortunately, decision analytical tools are not commonly used until now, despite their potential to really support rational decision making. Elaborated and sound models have been developed over the last 50 years, but a more substantial applicability of these is still limited coming to practical use. On the other hand, this is not entirely surprising considering the gap between the model's conceptual requirements and the actual background of the indented users in many cases; users who are often not capable of providing the input information that utility and MCDM theories require [17]. The idea behind this web application is therefore to provide a more easy and straightforward access to a systematized method for decision analysis with a user friendly interface. Furthermore, the application will be free for academic use and will successively be extended by knowledge data bases for specific domains and other support mechanisms for structured decision modelling processes, earlier reported in our works. The software is available at www.preference.bz.

REFERENCES

[1] R. Brase. (2010, 6 30). *Work with Java Web Start beyond the sandbox.* Retrieved from http://www.builderau.com.au/program/java/soa/Work-with-Java-Web-Start-beyond-the-sandbox/0,339024620,320269115,00.htm

[2] F. Burstein and C. W. Holsapple. *Handbook on Decision Support Systems 1*. Springer, 2008.

[3] Clemen, R. T. *Making Hard Decisions*. Duxbury Press, 1996.

[4] Creamtec. (2010, 6 30). *WebStart vs Java Plugin vs WebCream comparison - pros, cons and improvement.* Retrieved from http://www.creamtec.com/webcream/solutions/webstart_plugin_pros_cons.html

[5] CreamTec. (2010, 6 30). *Automatic Conversion of Swing Applications to AJAX websites*. Retrieved from http://creamtec.com/products/ajaxswing/index.html

[6] M. Danielson, L. Ekenberg, J. Johansson, and A. Larsson. "The DecideIT Decision Tool". *Proceedings of the Third International Symposium on Imprecise Probabilities and Their Applications*, 2003, pp. 204-217, Carleton Scientific.

[7] M. Danielson . "Generalized Evaluation in Decision Analysis". *European Journal of Operational Research* 162(2), 2005, pp. 442-449.

[8] M. Danielson and L. Ekenberg. "Computing Upper and Lower Bounds in Interval Decision Trees". *European Journal of Operational Research* 181(2), 2007, pp. 808-816.

[9] M. Danielson, L. Ekenberg, J. Idefeldt, and A. Larsson. "Using a Software Tool for Public Decision Analysis: The Case of Nacka Municipality". *Decision Analysis* 4 (2), 2007, pp. 76-90.

[10] M. Danielson, L. Ekenberg, A. Ekengren, T. Hökby and J. Lidén. "Decision Process Support for Participatory Democracy". *Journal of Multi-Criteria Decision Analysis* 15(1-2): 15-30, 2008.

[11] M. Danielson, L. Ekenberg, A. Larsson and M. Riabacke. "Transparent Public Decision Making –Discussion and Case Study in Sweden," *e-Democracy: A Group Decision and Negotiation Perspective*, Eds. Insua, D. R. and French S., Springer, 2010.

[12] Excelsior LLC. (2010, 6 30). *Desktop, Server, Embedded Java SE Implementations - Excelsior*. Retrieved from http://www.excelsior-usa.com/

[13] Hämäläinen, R., & Kalenius, R. (2010, 6 30). *Opinions-online—platform for global participation, voting, surveys, and group decisions*. Retrieved from Systems Analysis Laboratory, Helsinki University of Technology: http://www.opinions.hut.fi

[14] Kalinovsky, A. (2010, 6 30). *Java Developer Connection: Choosing Client-side Java Technology.* Retrieved from http://java.sun.com/developer/technicalArticles/GUI/frontends/

[15] R. L. Keeney and H. Raiffa. *Decisions with multiple objectives: Preferences and value tradeoffs*. John Wiley & Sons, 1976.

[16] R. L. Keeney. *Value-Focused Thinking*. Harvard University Press, 1992.

[17] R. L. Keeney. "Making Better Decision Makers". *Decision Analysis* 1(4), 2004, pp. 193-204.

[18] T. Könnölä, V. Brummer and A. Salo. "Diversity in foresight: Insights from the fostering of innovation ideas". *Technological Forecasting and Social Change* 74 (5), 2007, pp. 608-626.

[19] A. Larsson, J. Johansson, L. Ekenberg and M. Danielson. "Decision Analysis with Multiple Objectives in a Framework for Evaluating Imprecision. International Journal of Uncertainty". *Fuzziness and Knowledge-Based Systems* 13(5), 2005, pp. 495-510.

[20] A. Larsson, L. Ekenberg, Å. Wessel and H. Bach. "Non-Technical Survey: A Model for Evidence Based Assessment," to appear in The Journal of ERW and Mine Action, 2010.

[21] MinGW community. (2010, 6 30). *MinGW | Minimalistic GNU for Windows*. Retrieved from http://mingw.org/

[22] J. Mustajoki, R. P. Hämäläinen and M. Marttunen. "Participatory multicriteria decision analysis with Web-HIPRE: a case of lake regulation policy". *Environmental Modelling & Software* 19 (6), 2004, 537-547.

[23] Oracle Corporation. (2010, 6 30). *Java Plug-in Technology*. Retrieved from http://java.sun.com/products/plugin/

[24] Oracle Corporation. (2010, 6 30). *Java SE Desktop* Technologies. Retrieved from http://java.sun.com/javase/technologies/desktop/javawebstart/index.jsp

[25] Oracle Corporation. (2010, 6 30). *Welcome to NetBeans*. Retrieved from http://netbeans.org/

[26] Preference. (2010, 6 30). *Risk Analysis, Decision Analysis, Formal Methods, and Decision Tools*. Retrieved from www.preference.bz

[27] T. L. Saaty. *The Analytic Hierarchy Process*. McGraw-Hill Inc, 1980.

[28] Sun Microsystems. (2010, 6 30). *Overview.* Retrieved from http://java.sun.com/j2se/1.5.0/docs/guide/javaws/developersguide/overview.html

[29] Sun Microsystems. (2010, 6 30). *Deploying a Java Web Start Application.* Retrieved from http://java.sun.com/docs/books/tutorial/deployment/deploymentInDepth/createWebStartLaunchButtonFunction.html

[30] Thompson, A. (2010, 6 30). *JaNeLa - The Java Network Launch Analyzer*. Retrieved from http://pscode.org/janela/

[31] X. Zhu and A. P. Dale. "JavaAHP: a web-based decision analysis tool for natural resource". *Environmental Modelling & Software* 16 (3), 2001, pp. 251-262