The Facilitator - An Open Source Effort to Support Multiobjective Decision Making

Philip Heilman[a], Greg Davis[b], Paul Lawrence[c], Jerry L. Hatfield[d] and John Huddleston[e]

[a] *Southwest Watershed Research Center,USDA-ARS 2000 E. Allen Rd. Tucson, AZ 85719, USA (pheilman@tucson.ars.ag.gov)*

[b]*Netstorm, Brisbane, Queensland, Australia*

[c] *Queensland Department of Natural Resources and Mines, Brisbane, Queensland, Australia*

[d]*National Soil Tilth Laboratory, USDA-ARS, Ames, Iowa,USA*

[e]*Information Technology Center, USDA-NRCS, Fort Collins, Colorado, USA*

Abstract: Multiobjective Decision Support technology has matured to the point where new efforts should incorporate components of existing systems, rather than build everything from scratch. To help such efforts, the Java source code to a multiobjective decision support system called "The Facilitator" has been revised and released under the Mozilla Public License. This open source effort will be of interest to three groups. First, anyone needing a tool for multiobjective decision support can use the Facilitator. Users familiar with Java can now see how actions in the Facilitator are implemented or modify the user interface. Second, the open source approach might appeal to decision support problems that focus on quantifying management effects in a particular problem domain, rather than on developing another multiobjective decision component. Third, those with a different weighting or ranking algorithm for making multiobjective decisions could integrate that algorithm into the Facilitator. The community of researchers developing decision support tools for environmental problems is small relative to the magnitude of environmental problems, making cooperation essential.

Keywords: Decision Support System, Multiobjective Decision Making, Open Source

1. INTRODUCTION

The science used to manage natural resources has improved greatly over the past few decades, but there is still a fundamental problem integrating and applying our understanding of how management affects natural systems. Because of the first rule of ecology, which is variously described as "you can't change just one thing" or that "everything is connected", any human modification of a natural system will affect other components of the natural system and ultimately the society that interacts with it. Computer programs that describe these multiple effects and provide a structured approach to selecting a management plan based on an individual's or groups' preferences and tradeoffs are called multiobjective decision support systems (MODSS).

Many such MODSS and associated processes for engaging stakeholders have been developed and described in El-Swaify and Yakowitz (1998) and Lawrence et al. (2002). MODSS are clearly an improvement over approaches that simply quantify or optimize one variable while ignoring other variables of interest. Nevertheless, given the scope and complexity of the inherently multiobjective environmental problems facing the world, MODSS are far from achieving their potential.

The field of multiobjective decision support for environmental management has matured to the point where efforts to develop MODSS for a particular problem should build on the strengths and contributions of existing software implementations without redundantly building all components from scratch. Rather, efforts to develop a new MODSS should only be required to contribute the new components needed to address the particular multiobjective problem of interest. Ideally, those new components would also be made available in such a way that later MODSS

efforts could incorporate those new contributions. The purpose of this paper is to announce the availability of a generic multiobjective decision support system, called *The Facilitator*, including the source code, and to present enough information about this open source effort so that interested users can assess its potential to complement their MODSS development efforts.

2. BACKGROUND AND DESCRIPTION

Wymore (1988) proposed the basic multiobjective decision-making approach in the Facilitator, which was subsequently adapted to natural resource decision-making in Lane et al. (1991) and implemented as the Water Quality Decision Support System, WQDSS (SWRC, 1994). Principal components of the WQDSS were a modified CREAMS model to quantify the management impacts on decision criteria, scoring functions that reduced the impacts to dimensionless values relative to the current management practice, and a range of weight method (Yakowitz et al., 1993) to rank alternatives. An application of this method to water quality problems in agriculture is described in Yakowitz et al. (1992). The WQDSS has also been used for other applications including shallow land burial systems for low-level nuclear waste (Paige et al., 1996), targeting farms for planning (Heilman et al., 1997) and rangeland planning (Lawrence et al., 1997). Imam (1994) addressed modeling and uncertainty issues.

To advance the WQDSS beyond the Unix environment and remove the overhead of a field scale simulation model, the Queensland Department of Natural Resources and Mines (QNR&M) in association with the National Heritage Trust in Australia contracted with Netstorm, Inc. to implement the decision-making component of the WQDSS in the multi-platform Java language (Lawrence and Shaw, 2002). The new software is a generic, multiobjective decision-making tool called the Facilitator, and incorporates the hierarchy tree of decision criteria by Yakowitz and Weltz (1998) and utilizes information from a range of sources to build the effects matrix that quantify the impacts of the options on each decision criterion. The development of the Facilitator was undertaken within an International Memorandum of Understanding between the USDA-ARS and the QNR&M.

Applications of the Facilitator have focused on planning for water infrastructure development in Queensland (Lawrence et al., 2000; Robinson et al, 1999), although the Facilitator has also been used to evaluate farming systems, floodplain management, farm forestry, animal production, project evaluation, and regional and watershed community strategy prioritizations. The Facilitator was designed for making strategic decisions where the problems are complicated enough to require a structured approach, and technical support is available to follow up on key issues affecting the decision.

3. THE ALGORITHM

The three basic steps to make a decision using the Facilitator are: 1.) create a matrix of the effects of each alternative on each criterion by defining the decision variables or criteria, the management alternatives to be considered, and quantifying the effects of the alternatives on the criteria; 2.) score all values in the matrix to eliminate units and normalize elements to a scale of 0.0 to 1.0, with 1.0 being as good as possible; 3.) rank the decision variables in order of importance, graphically examine the results, and select the alternative(s) to implement or for more detailed investigation. Lawrence et al. (2001) describe a richer conceptual framework of the many considerations and processes that lead to the definition of the alternatives and decision criteria within the decision analysis.

When performing the first step, decision-makers are responsible for excluding alternatives that are considered to be unacceptable. In the second step, decision-makers select score functions for each decision variable from among the following choices: more is worse, more is better, a desirable range, or an undesirable range. The "more is worse" score function is used for a decision variable like the quantity of pollutants leaving a field or rise of groundwater levels, while net returns or grain yield are examples of decision variables that would be scored using a "more is better" score function. The sigmoid or bell-shaped functions used to score variables are described in Wymore (1988, p. 707), although the Facilitator also provides linear score functions.

The third step assumes a simple additive value function of the form:

$$V(w,v) = \sum_i w_i v_i \qquad (1)$$

to calculate an overall value, V, as the sum of the products of a weight, w, associated with each decision variable, or criterion, i, and the score, v, for that decision variable. Although conceptually simple, the approach can be difficult to apply in practice because decision-makers find it difficult to assign weights. Yakowitz et al. (1993)

developed a method that eliminates the need for decision-makers to specify a weight for each decision variable. Instead, the decision-makers

$$v_{kj} = \frac{1}{k} \sum_{i=1}^{n} v_{ij} \qquad (3)$$

Table 1. Matrix of estimated effects (and scores) for example problem.

	Cost	Off-Watershed Effects	Upstream Stakeholders	Downstream Stakeholders
No MODSS	1.0	0.2	0.8	0.2
Simple MODSS	0.8	0.4	0.6	0.5
Facilitator	0.5	0.5	0.4	0.6
Complex MODSS	0.1	0.6	0.2	0.7

rank the decision variables in order of importance that assigns all possible weighting combinations to the decision variables. A cost of using the method is that a range of values representing the overall value for the alternative is calculated, rather than a scalar value that quantifies the overall value of a particular alternative.

The method developed by Yakowitz has an intuitive appeal to decision makers. Suppose there are n criteria, which the decision-maker has ranked in importance. Let V_{ij} be the score of alternative j evaluated with respect to criterion i in the importance order. If w_i indicates the unknown weight factor associated with criterion i, the highest (lowest) or best (worst) additive composite score for alternative j, consistent with the importance order, is found by solving the following linear program described for the weights w_i:

$$max(min) \ V_j = \sum_{i=1}^{n} w_i v_{ij} \qquad (2)$$

$$subject \ to \ \sum_{i=1}^{n} w_i = 1$$

$$w_1 \geq w_2 \geq \ldots \geq w_n \geq 0.$$

In both cases (maximizing or minimizing) the first constraint normalizes the sum of the weights to 1, while the second requires that the solution be consistent with the importance order and restricts the weights to be nonnegative. The solution to the two programs yields the full range of possible composite scores given the importance order. Any weight vector that is consistent with the importance order will produce a composite score that falls between the best and worst composite scores.
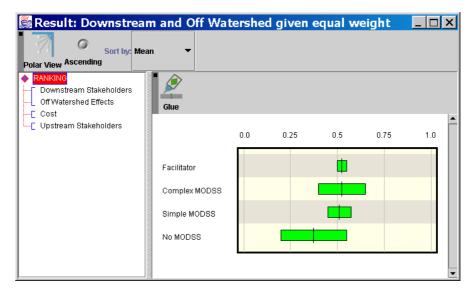
Yakowitz et al. (1993) also showed that the best and worst composite scores could be calculated in closed form, as the maximum or minimum composite score could be calculated by solving the following k problems, starting at the highest ranked criterion and adding criteria until they have all been considered:

The best or worst composite score for alternative j is then selected from the results as:

$$BestScore = BV_j = \max_k \{v_{kj}\},$$

$$WorstScore = WV_j = \min_k \{v_{kj}\} \qquad (4)$$

A later study (Yakowitz and Weltz, 1998) improved the weighting algorithm by incorporating a hierarchical importance ordering, so that a number of sub-objectives could be grouped under categories such as "water pollutants" or "sustainability". The hierarchy approach also addressed issues of bias caused by having too many criteria of one type (for example, environmental) compared to other considerations (for example, economic, social, cultural).

Detailed examples of decision-making using this approach, either in the WQDSS or the Facilitator, can be found in the previously cited papers. A simple, illustrative example of the data needed to address a watershed management problem with the Facilitator is shown in Table 1. The current situation, labelled "No MODSS", is that upstream stakeholders are polluting, affecting downstream stakeholders within the watershed, as well as others downstream of the watershed outlet. This state of affairs has no expenditures, and so the score for the cost variable is 1.0, as good as possible. Efforts to implement MODSS approaches of various levels of complexity will have costs proportional to their complexity and thus have lower scores for the cost criterion. For simplicity, all of the other possible decision variables have been aggregated into the other three decision variables representing the groups involved. It is assumed that the more complex the approach, the greater the pollution reduction, and consequently the greater the benefit to the downstream and offsite stakeholders, with a concomitant increased loss to the upstream stakeholders. In this case, the experts are assumed to directly estimate scores for each alternative and criteria, so the raw data matrix and the score matrix are identical.

Figure 1. A sample screen of the Facilitator

If downstream and off-watershed stakeholders are equally ranked and both are more important than the costs or the upstream stakeholders, the graphical results would be shown as in Figure 1. Additional analysis with other rankings and assessments of the effect of individual decision variables could also be done in the Facilitator. The bars contain all possible overall scores for each alternative consistent with the given importance order. In this case, the three options using a MODSS are preferred to the No MODSS option for almost all weightings.

The choice between the three MODSS options would ultimately depend on the decision-makers' tradeoff between cost and the downstream and off-watershed stakeholders. If cost were given as much weight as possible, the Simple MODSS would be preferred. If decreasing the effect of downstream pollution were given extra weight, then the Complex MODSS would be preferred, with the Facilitator preferred for the intermediate case.

4. ACCESSING THE SOURCE CODE

4.1 An Invitation

Programmers or developers of MODSS have access to the entire Facilitator package via an open source repository known as SourceForge. This includes Java source code, resources (including images) and documentation. The Facilitator can be checked out from http://facilitator.sourceforge.net/ using the Concurrent Versions System (CVS). Documentation describes the organization of the package structure. The code for Facilitator was designed and written in 1997 as a standalone MODSS system. Since then it has gone through a number of upgrades to reach its current form. A mailing list is used for communication between programmers/developers.

Currently the Facilitator runs on any platform that supports the Java Development Kit 1.3 or greater. A build/deploy process is provided based on the Ant Java build tool. In late 2001, a significant amount of work went in to refactoring (improving the design without changing the functionality) the Facilitator source code. The core model has been completely rewritten, and now lends itself to evolution. The refactoring work saw the external representation of Facilitator data move from Java serialised objects to the Extensible Markup Language (XML). It is anticipated that the wide and varied needs of MODSS users will drive the Facilitator development to a point where it can fit easily into a variety of systems, becoming a truely reusable software component. Of course, the code is available on an "as is" basis, and there is no warranty expressed or implied.

4.2 License restrictions on the Facilitator

Open source software is sometimes confused with public domain software. Code released to the public domain is available with no restrictions, whereas code released as *open source* is usually released under a license that restricts, in some measure, the use that can be made of that software. The source code to the Facilitator is offered under the Mozilla Public License Version (MPL) 1.1.
The license gives the rights to view, use, modify, and redistribute the source code for free. Unlike the Gnu General Public License, the MPL is not "viral" in that the Facilitator code could be used as part of a larger decision support system and the developers could keep the additional, non-Facilitator, code proprietary. Similar to the Gnu

General Public License, developers can charge for projects incorporating MPL code. For a complete list of the license restrictions see the MPL. Some of the major restrictions in the MPL are that a copy of the License must be included with every copy of the source code distributed and any modification to the source code must be made available in source code form under the terms of the MPL.

5. POTENTIAL APPLICATIONS

5.1 Applications that use the source code

The source code to the Facilitator could be put to a number of uses. For those who know the Java programming language, it is possible to examine the code to see how steps in the decision making process are implemented. To convert the Graphical User Interface to a non-English language, or to customize the wording (for example, "options" rather than "alternatives") the only change needed is to edit a properties file.

Perhaps the most likely application of the source code for the Facilitator is to support efforts that focus on understanding and modeling the physical or biological processes in a particular problem domain, but that have not yet incorporated any decision making framework. For such efforts, it is straightforward to use the modeling efforts to quantify the expected effects of different management scenarios, and then use that information directly in the Facilitator.

5.2 Contributing to the source code

Research efforts focused on multiobjective decision-making are more likely to want to modify, and contribute to, the Facilitator source code. There are a number of ways the Facilitator could be improved or enhanced. The interface could be improved to allow more flexibility for advanced users and provide a graphical presentation of the progress toward reaching a decision. Enhancements to allow the Facilitator to be used over the Web would be useful, as often groups who need to work together are geographically dispersed. Support for groups to negotiate to agreement on a mutually acceptable alternative could be included.
Improvements in the decision-making process used by the Facilitator are also possible. One possibility is to include other multiobjective decision-making techniques as options, particularly where these approaches are conducive to sensitivity analyses. Another is to provide additional support for defining score functions. Methods to provide the user with a better

understanding of the uncertainty associated with each decision would be a useful contribution, as would methods to improve the ability to discriminate between alternatives. Significant changes to the design and implementation of the Facilitator will need coordination and adjustment to the user's manual.

6. DISCUSSION

There is a long history of researchers sharing ideas and data to improve our understanding of the natural world and to develop better tools to achieve human goals within nature's constraints. Carefully designed software made available through the Web can facilitate a more structured and coordinated approach to improving environmental management, where literally anyone in the world can apply, and contribute to, common efforts to develop better tools to manage natural resources.

Obviously, the Facilitator is not necessarily the only approach for all multiobjective issues. If there are efforts that are better suited to simpler, or more complex problems, or that use a different approach, those efforts should also be made widely available, ideally with open source code. Although the Facilitator is currently the focus of the website www.modss.org, that site is dedicated to the promotion of multiobjective decision support systems as a whole and improving communication among the MODSS community, so other MODSS efforts could also be featured.

We encourage others to provide open source code that could improve multiobjective decision-making on natural resource management issues. Good candidates for open source efforts would have the following characteristics:
- Address a widespread, significant problem
- Have been sufficiently validated and used to minimize software bugs
- Not contain proprietary code
- Have source code that is well documented and easy to follow

In summary, MODSS can provide a structured approach to engage stakeholders in complex environmental issues where there are many, and possibly conflicting, criteria to consider. Many of the applications of MODSS, such as watershed management, will contain common features no matter where in the world they are applied. We offer the source code to a generic MODSS, the Facilitator, to any group to use, and we hope those groups will apply it and collaborate to improve

upon the initial developments. We also invite other MODSS applications for environmental and resource allocation problems to publicize their outcomes on the MODSS website. Our hope is that other groups will join with us to work on distributed, cooperative efforts to improve MODSS that will ultimately lead to more sophisticated, informed and complete resolution of environmental problems.

8. ACKNOWLEDGEMENTS

9. REFERENCES

El-Swaify, S.A., and D. S. Yakowitz, eds. Multiple Objective Decision Making for Land, Water, and Environmental Management: Proceedings of the First International Conference on Multiple Objective Decision Support Systems Lewis Publishers, New York, 1998.

Heilman, P., D.S. Yakowitz, L.J. Lane, Targeting farms to improve water quality, *Applied Mathematics and Computation* 83:173-194, 1997.

Imam, B., Nonlinear Uncertainty Analysis of Multiple Criteria Natural Resource Decision Support Systems, Ph.D. Dissertation, University of Arizona, Tucson, 1994.

Lane, L.J., J.C. Ascough, and T. Hakonson, Multiobjective Decision Theory - Decision Support Systems with Embedded Simulation Models. Irrigation and Drainage Proceedings 1991, Honolulu, HI, July 22-26, 1991.

Lawrence, P.A., and J. Robinson, Multiple Objective Decision Making for Land, Water, and Environmental Management: Proceedings of the Second International Conference on Multiple Objective Decision Support Systems (MODSS), Queensland Department of Natural Resources and Mines, 2002.

Lawrence, P. A., J. Robinson, J. and R. Eisner, A decision environment: going beyond a decision framework to improve the effectiveness of decision making in natural resource management, International Congress on Modelling and Simulation (MODSIM 2001) Canberra, Australia: 1613-1618. 2001.

Lawrence, P.A., R.J. Shaw, L.J. Lane, L.J. and R. Eisner, Participatory Multiple Objective Decision Making Processes: Emerging approaches with new challenges. Proceedings Watershed Management and Operations Management 2000, American Society Civil Engineers, 11 pp. 2000.

Lawrence, P.A., J.J. Stone, P. Heilman, and L.J. Lane, Using Measured Data and Expert Opinion in a Multiple Objective Decision Support System for Semiarid Rangelands, *Transactions of the ASAE*, Vol. 40(6):1589-1597, 1997.

Paige, G.B, J.J. Stone, L.J. Lane, D.S. Yakowitz, and T.E. Hakonson, Evaluation of a Prototype Decision Support System for Selecting Trench Cap Designs, *J. Env. Qual*, 25:127-135, 1996.

Robinson, J., P. A. Lawrence, R.J. Shaw, L. Cogle, K. Rose, and R. Lait, Development and outcomes from a multiple objective decision support system for the Cattle Creek Catchment, Department of Natural Resources report, 1999.

Southwest Watershed Research Center, A Multiple Objective Decision Support System for the USDA Water Quality Initiative: Reference Manual, Version 1.1 , 1994.

Wymore, A.W., Structuring system design decisions. In: Systems Science and Engineering (Ed. C. Weimin), Proc. Intrl. Conf. on Systems Science and Engineering (ICSSE), Beijing, China. pp. 704-709, 1988.

Yakowitz, D.S., L.J. Lane, J.J. Stone, P. Heilman, R.K. Reddy, A Decision Support System for Water Quality Modeling, Water Resources Planning & Management Proceedings of the Water Resources Sessions/Water Forum, WR Div/ASCE, Baltimore, MD, 1992.

Yakowitz, D.S., L.J. Lane, and F. Szidarovsky, Multi-attribute decision making: dominance with respect to an importance order of attributes, *Applied Mathematics and Computation*, Vol. 54, pp. 167-181, 1993.

Yakowitz, D.S. and M. Weltz, An algorithm for computing multiple attribute additive value measurement ranges under a hierarchy of the criteria: application to farm or rangeland management decisions, in *Multicriteria Analysis of Land-Use Management*, E. Beinat and P. Nijkamp, eds., Kluwer Academic Publishers, Boston, pp. 163-177, 1998.