Multi-objective Decision Support Including Sensitivity Analysis*

Marek Makowski
International Institute for Applied Systems Analysis,
Laxenburg, A-2361 Austria
`http://www.iiasa.ac.at/~marek`
`e-mail:marek@iiasa.ac.at`

Contents

*This is a draft version of the article accepted for the *Encyclopedia of Life Support Systems*.

Keywords: decision making, decision support systems, multiple-criterion optimization, mathematical modeling, linear programming, nonlinear programming, robustness, simulation, optimization, Pareto-optimality, criterion functions, efficient solutions, sensitivity analysis, model generation, model analysis, natural environment, air quality, water quality.

Summary

The article presents methodological background of multi-objective decision support and its applications to environmental decision-making problems. It starts with discussing basic concepts of model-based decision support by outlining the characteristics of modern decision makers and presenting in more detail possibilities and limitations of operational research methods to providing useful support for actual decision making processes. In particular, methods of formulation and analysis of mathematical models that allow for their efficient use for decision support are described.

The article concentrates on one of the most efficient way of model analysis for decision-making support, namely on multi-objective model analysis methods. These methods are summarized and illustrated by a software tool, which has been widely used in various environmental applications of actual decision making support. The relations between these methods and two basic classical methods of model analysis, namely simulation and single-objective optimization are discussed. One particular method of multi-objective model analysis, namely aspiration-reservation based model analysis is discussed in more detail and illustrated by an outline of its implementation. Presentation of multi-objective model analysis methods finishes with an overview of methods specialized for multi-criteria analysis of a small set of alternatives.

The article outlines also a general structure of model-based decision support systems, which is based on actual implementations of the presented methods to real-world environmental policy problems of various scales applied to several problems, including the quality of air and water, and to land use planning.

Finally, advantages and limitations of model-based decision support are briefly summarized.

Glossary

ARBDS Aspiration-Reservation Based Decision Support: methodology and tools for multi-criteria model analysis, which is based of specification of user preferences by definition of aspiration and reservation levels.

Aspiration level: Value of a criterion that the user wants to achieve.

The Analytic Hierarchy Process (AHP): The method of analysis of a finite set of alternatives. The alternatives are examined in pairs and their relative performance in evaluated as a ratio of subjective values or as a difference of grades.

Component achievement function (also called partial achievement function): a scalarizing function, which reflects the level of satisfaction of the user for each value of a criterion.

Compromise solution: Pareto-optimal solution obtained for aspiration and reservation levels set to the Utopia and Nadir points, respectively.

Decision Maker (DM) A person, who is responsible for making actual decision. Often this term is used also for experts, advisors, analysts, even researchers, in other words anybody, who uses analytical methods for decision analysis.

Decision Support System (DSS): Specialized software used for analysis of models aimed at supporting decision-making process.

Habitual Domain (HD): A fairly stable set of ways for thinking, evaluating, judging and making decisions.

LP Linear Programming: the type of mathematical programming problem, in which the goal function and all constraints are linear.

Multi-Criteria Model Analysis (MCMA): Methodology and corresponding software tools aimed at supporting decision-making by developing and analysis of mathematical models. The analysis supports examination of these parts of Pareto-set that best correspond to preferences of the DM, which are interactively specified by a sequence of selections of aspiration and reservation levels.

MIP Mixed Integer Problem: an LP problem, in which a subset of variables can take only integer values.

Nadir point: A point in the criteria (objective) space, composed of the worst values (from the Pareto-set) of all criteria.

NLP Nonlinear Programming: the type of mathematical programming problem, in which the goal function and/or some of constraints are nonlinear.

OR: Operations Research (a.k.a. Operational Research): a set of quantitative methods and tools developed to assist analysts and decision-makers in designing, analyzing, and improving performance or operation of systems.

Piece-Wise Linear Function (PWL): a continuous function that is linear except of a finite set of points (where its slope changes). Often used for approximation of non-linear functions in order to make it possible to deal with LP or MIP problem instead of NLP problem.

Pareto-optimal solution: A solution of a multi-criteria problem for which there exists no other solution for which at least one criterion has a better value while values of remaining criteria are the same or better.

Pareto set: A set of all Pareto-optimal solution. Typically, except of problems of finite number of alternatives, there is an infinite number of Pareto-optimal solutions.

Reservation level: Value of a criterion that the user wants to avoid.

Simple Multi-Attribute Rating Technique (SMART): The method of analysis of a finite set of alternatives. The performance of each alternative is expressed in grades on a numerical scale, which are evaluated through a direct-rating procedure.

Utopia point: A point in the criteria (objective) space, composed of the best values of all criteria.

Information about the author

Sometime ago I've been asked by Ms Eibl to provide some information about myself. Below please find a collection of relevant facts, please select from the information that fits to the requirements of EOLSS.

Marek Makowski is a senior research scholar of the International Institute for Applied System Analysis (IIASA), Laxenburg, Austria. Before joining IIASA in 1987 he had been working, since 1970, at the Systems Research Institute of the Polish Academy of Sciences, Warsaw, Poland, since 1974 as a head a laboratory.

His main research topics: decision support type applications, development of optimization solvers and software for multiple-criteria model analysis. Research interests include applications of mathematical programming methods and of relevant user interfaces in decision support systems, the development of methodology, algorithms and software for model-based decision-making support using multicriteria model analysis methods.

He has been either leading or involved in several projects aimed at the development of decision support systems that has been used for various complex decision and policy problems, including modeling of the Polish agriculture, design of water system in the Upper Notec region, regional water quality management in Slovakia, land-use planning in several countries, European air quality model used in the intergovernmental negotiations.

His academic education started in 1964 at the Faculty of Electronics, the Warsaw University of Technology. In 1966-1969 studied also applied mathematics at the Faculty of Mathematics and Mechanics, the Warsaw University. Graduated (MS) in 1970, in the field of control engineering and computer sciences. Ph.D. received his 1976 from the Systems Research Institute of the Polish Academy of Sciences for a thesis on optimization of environmental models.

1 Introduction

The complexity of environmental problems requiring rational decision making and of the decision making process have been rapidly growing. Globalization, interlinks between environmental, industrial, social and political issues, and rapid speed of change all contribute to the increase of this complexity. While decision-making is becoming more and more difficult, especially for environmental problems, there are methodologies and tools which – when used properly – can greatly assist modern decision makers in making better decisions.

However, these methods and tools are not available in a *ready from the shelf* form that can easily be adopted for supporting decision making in complex problems. They are rather components, which can be used by skilled teams of modelers, who can develop in a close collaboration with future users, a problem specific Decision Support System (DSS). A skilled team of modelers is needed because heterogeneous knowledge about decision-making processes and their support is rapidly increasing, and therefore a rational selection of methods and tools that are appropriate for a specific problem requires expertise in several fields. Moreover, a large spectrum of approaches presented in the literature is typically illustrated only by simple examples, and the range of their applicability is often exaggerated. Therefore, integration of model-based decision support methodologies and tools with specialized model-based knowledge developed for handling real environmental problems which typically need to be considered together with the corresponding engineering, industrial, economical, social and political activities requires various skills of an interdisciplinary team.

Environmental decision problems always require multiobjective approach because they typically involve analysis of trade-offs between conflicting objectives, such as various costs and indicators of the state of environment. Depending on the type of the decision problem, different methods of multicriteria problem analysis are appropriate from the methodological point of view. However, the habitual domain of a Decision Maker (DM) is a far more important factor for the selection of methods than their theoretical correctness. In reality, the decisions are made by DMs, and not by the developers of DSSs. Therefore a DSS will be used only if the implemented methods and assumptions will be understood and accepted by those who actually make the decisions and take responsibility for consequences of their implementations. Moreover, the developers of a DSS have to understand the decision making process and its environment, which is always specific for each problem. This is why close collaboration between the developers of a DSS and a DM is one of the necessary conditions for a proper design and implementation of each DSS. It also justifies the observation that no *ready from the shelf* tool can be actually applied to supporting decision making in a complex problem.

This article starts with a summary of basic concepts of model based decision support. Then it presents in more detail methods and tools for multiobjective model analysis for decision support. Finally the advantages and limitations of model-based decision support are summarized.

2 Basic Concepts

2.1 Modern DM

A selection of methods and tools for supporting decision making is to a large extend determined by a characteristic of persons, who need and want to use such support. Therefore, one should start with an outline of such a characteristic.

For the sake of brevity, by Decision Maker (DM) we understand here not only anybody, who actually makes decisions or takes part is a Decision-Making Process (DMP) but also experts, advisors, analysts, even researchers, in other words anybody, who uses analytical methods for decision analysis. The word *Modern* stresses two facts: First, as already discussed in the Introduction, that rational decision making is becoming more and more difficult; second, that the developments in decision support methods and tools offer nowadays help for comprehensive analysis of various aspects of the problem at hand, including examination of possible outcomes of various decisions and identification of decisions that correspond best to a preferential structure of a DM.

Any DM wants to understand in a best possible way the consequences of implementations of his/her decisions before making the final selection of a decision. Moreover, especially in more complex situations, a DM typically needs help in finding decisions that correspond best to her/his preferences. The preferences can hardly be precisely defined in advance because they often change while a DM learns about the decision problem. Experiences show that such learning is an important element of the development and use of a DSS for any complex problem, for which intuition and expertise of a DM are not enough for predicting consequences of various decisions. Finally, a DM typically also wants to examine consequences of decisions that he/she defines, often by using own intuition and/or experience for modifications of decisions obtained from other analysis or suggested by somebody else.

A modern DM is confronted with more complex decision problems than previous generations of DMs, but she/he has much better knowledge about decision making processes and access to analytical tools and teams of experts and advisors. Therefore, such a DM is not keen to accept the classical OR approach based on using a given solution of a mathematical model which represents a well-structured problem as the basis of a decision. He/she needs a DSS which can be used for various types of analysis which can help to extend the DM's knowledge about the problem and allow to take advantage of his/her experience and intuition. In order to achieve this, a good DSS can be considered as being composed of two mutually linked parts that are of different nature:

- A mathematical model that represents a part of DMP for which logical and physical relations that have to be considered for a given decision problem but which should be handled in a form of a model rather then by intuition or experience of a DM. Such a model is further on called a *core model*.
- Tools for a comprehensive analysis of a core model.

2.2 Core model

Mathematical and computer models are widely used in many areas of science and industry for predicting the behavior of a system under particular circumstances, when it is undesirable or impossible to experiment with the system itself. The understanding of the system gained through a comprehensive examination of its model can greatly help in finding decisions (controls) whose implementations will result is a desired behavior of the system.

The systems that are modeled have very different characteristics (including the nature of the underlying physical and/or economical processes, their complexity, size, types of relations between variables). There is also a great variation in the use of models, which depends on various factors (like the decision making process, the background and experience of model users). However, the modeling process (composed of problem formulation, model specification, implementation, verification and validation, analysis and management) has many similarities also when the modeled systems are very different. The modeling process is a combination of craftsmanship and art, and its quality is critical for any model-based DSS. However, a discussion of the related issues is far beyond the scope of this article, therefore a reader interested in these issues is advised to consult some of the references listed at the end of this article.

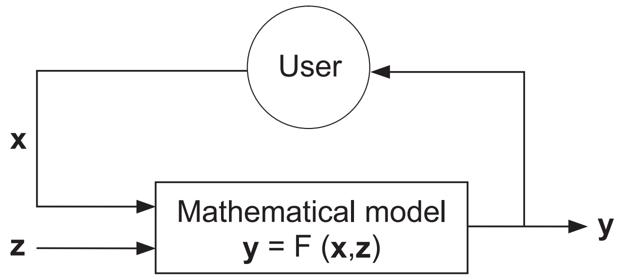

Figure 1: A mathematical model represents relations between decisions (inputs) **x**, external decisions (inputs not controlled by the user) **z**, and consequences (outcomes) **y**.

Only basic concepts and features of a core model, which are essential for explanation of model-based decision support are discussed here. The basic function of a core model is to provide an evaluation of consequences that will result from an implementation of given decisions. All four basic concepts illustrated on Figure 1, namely decision variables, external decisions, outcome variables and a mathematical model are briefly discussed in the following subsections.

2.2.1 Illustrative examples

In order to illustrate several concepts discussed in this article we outline here two actual DSSs, which are presented in more detail in article 4.20.4.2.

The first model, referred to as RWQM (from Regional Water Quality Management), is applied to a region, in which untreated or inadequately treated municipal and industrial wastewater emissions should be reduced in order to improve ambient water quality. At each discharge point, one technology out of a set of possible technologies can be implemented in order to meet the desired water quality goals in the region. In this selection of technologies, or strategy development, decision makers must evaluate the trade-offs among a large number of alternatives based on, among other things, effluent and/or ambient water quality standards and goals, as well as capital investment and annual operating costs.

The second model, RAINS provides a consistent framework for the analysis of emission reduction strategies, focusing on acidification, eutrophication, and tropospheric ozone. RAINS comprises modules for emission generation (with databases on current and future economic activities, energy consumption levels, fuel characteristics, etc.), for emission control options and costs, for atmospheric dispersion of pollutants, and for environmental sensitivities (i.e., databases on critical loads). In order to create a consistent and comprehensive picture of the options for simultaneously addressing the three environmental problems (acidification, eutrophication, and tropospheric ozone), the model considers emissions of SO_2, NO_x, ammonia (NH_3), and VOC.

RWQM is a rather small (composed of about 1000 variables) MIP type model, and RAINS is a large (about 30000 variables) non-linear model. Both of them have been used for supporting complex decision making processes, and they illustrate well various issues of model-based DSS.

2.2.2 Decision variables

In model-based decision support it is assumed that decisions have quantitative characters and therefore can be represented by a set of the model variables, hereafter referred to as decisions[1] $x \in E_x$, where E_x denotes a space of decisions. In a trivial case $x \in R$, which denotes that a decision is represented by a real number. However, in most cases x is a vector composed of various types of variables. For larger problems the components of x are aggregated in several vectors. Let us illustrate this by specification of the decision variables of our illustrative models.

In the RWQM model the decision variables are the treatment technologies to be implemented at the nodes where waste-water emissions occur. A selection of a technology is represented by a binary variable x_{jk} (which takes value of 1, if a corresponding technology is selected, and value of 0, otherwise), k is the technology choice at emission node j. These technology options include the option of no treatment (with raw waste concentrations and no cost, which is actually only a theoretical alternative since minimum treatment levels would likely be required), as well as the option of maintaining the existing technology (with the operating cost but no investment cost). Therefore, formally, the decision vector x is composed of vectors x_j, each of which is a vector of binary variables. This convention is of course used only for modelers. Results are provided for users of the RWQM with a help of a mapping, which shows a description of a selected technology for each location.

In the RAINS model the main decision variables are the annual emissions of the following four types of primary air pollutants from either sectors or countries:
- n_{is}, annual emission of NO_x from sector is;
- v_{is}, annual emission of nonmethane VOC from sector is;
- a_i, annual emission of NH_3 from country i; and
- s_i, annual emission of SO_2 from country i.

where sectors is are grouped for each country.

Additionally, optional decision variables are considered for scenarios that allow limited violations of air quality targets. For such scenarios variables corresponding to each type of considered air quality targets are defined for each receptor. Each variable represents a violation of a given environmental standard. Optionally, violations of targets can be balanced with surpluses (understood as difference between a target and its corresponding actual concentration/deposition).

2.2.3 External decisions

Figure 1 illustrates two types of inputs to the core model: decision variables $\mathbf{x}$ controlled by a user, and external decisions denoted by $\mathbf{z}$. In practice inputs $\mathbf{z}$ may include representations of various quantities that substantially influence the values of outcomes $\mathbf{y}$ but are not controlled by the user, for example:
- Regulations or commitments on environmental standards for air or water quality management models.
- Meteorological conditions assumed for modeling physical relations in environmental models, e.g. *average, wet, dry, worst* year data for a water model.

[1] For the sake of brevity we call decision variables simply decisions.

- Forecasts of changes in demand for services, e.g. in telecommunication or transportation models.

In the RWQM and RAINS models the external decisions $\mathbf{z}$ are represented by:

- Values representing the environmental standards that define constraints for various indices (such as maximum concentrations of various water and air quality indicators, respectively).
- Set of meteorological data used for calibration of a respective model.

While the external decisions are beyond control of the user of a DSS, he/she typically wants to examine a range of scenarios with various representations of external decisions in order to find out not only a solution which will best respond to a most likely representation of external inputs $\mathbf{z}$ but also a solution that will be *robust*, i.e. will be good also for various other compositions of $\mathbf{z}$ that should be considered.

2.2.4 Outcome variables

The consequences of implementation of various decisions x are evaluated by values of outcome variables $y \in E_y$. In various fields of applications outcome variables are named differently, e.g. outcomes, metrics, goals, objectives, performance indices, attributes.

In RWQM there are two sets of outcome variables:

- Three types of costs (investment, operating and maintenance, and total annual cost) related to the implementation and operation of a given selection of water treatment technologies.
- Several types of water quality indicators, such as an extremum (over the set of monitoring points) of concentrations of CBOD (carbonaceous biochemical oxygen demand), NBOD (nitrogenous biochemical oxygen demand) NH4 (ammonia), DO (dissolved oxygen); depending on the type of a water quality constituent, as the extremum either minimum (e.g. for DO) or maximum (for NBOD, CBOD and NH4) are taken.

In the RAINS model one outcome variable represents the sum of costs of reductions of emissions; four sets of additional outcome variables correspond to various indices of air quality. While the definition of the cost is rather simple, an appropriate definition of air quality indices is rather complex. Environmental effects caused by acid deposition, excess nitrogen deposition (described by a two-element linear critical loads function), and by eutrophication are evaluated at each receptor by a PWL (piece-wise linear) function that represents an accumulated excess over the threshold of the environmental long-term target. If optional violations of environment standards are allowed, then a maximum (over a set of receptors) violation of each type of air quality indicator is also considered as an output variable.

2.2.5 Objectives

Out of the set of outcome variables $y \in E_y$, a user can select a subset of objectives $q \in E_q$, where E_q is a space of objectives. Quite often objectives are referred to as criteria, and in this article these two terms will be used interchangeably.

Usually E_q is a subspace of E_y, that is, the DM select some criteria q_i from the set of outcomes y_j. Sometimes also some of the decision variables x are used as criteria, but for the sake of consistency we assume that such a variable is simply represented by one of the outcomes y. Such a set of objectives is typically modified during model analysis.

A partial preordering in E_q is usually implied by the decision problem and has obvious interpretations, such as the minimization of costs competing with the minimization of pollution. However, a complete preordering in E_q cannot usually be given within the context of a mathematical programming model. In other words, it is easy to determine for each objective separately, which solution (represented by vectors x and q) is the best one. However, for conflicting objectives there are two sets of solutions:

- Pareto-optimal (often called efficient), i.e. a solution, for which there is no other solution for which at least one criterion has a better value while values of remaining criteria are the same or better.
- Dominated, i.e. solutions, which are not Pareto-optimal.

Obviously, a Pareto-optimal solution is preferred over any dominated solution (assuming, that the selected criteria correspond well to the preferential structure of a DM). However, a set of Pareto-optimal solutions (often called Pareto-set, or Pareto frontier) is typically composed of a infinite number of solutions, many of which are very different. Pareto-optimal solutions are not comparable in a mathematical programming sense, i.e. one can not formally decide which is better than another one.

However, DMs are able to express own preferences for various efficient solutions. One of the basic functions of multiobjective decision support is to provide various ways in a DM may specify his/her preferences. There is no reliable formal way for separating a specification of preferences from a process of learning from the model analysis. It is a commonly known fact that decision making is not a point event, even in situations where it is realistic to assume that the problem perception does not change during the DMP. Therefore, the possibility of using a DSS in a learning and adaptive mode is a critical feature.

2.2.6 Mathematical model

As already illustrated in Figure 1 on page 7, a mathematical model is used for predicting the consequences of decisions x, which can be either proposed by a DM or computed by a DSS. The consequences are measured by values of outcome variables y. Therefore, a model can be represented by mapping $y = F(x, z)$, where $x \in E_x$, $z \in E_z$, and $y \in E_y$ are vectors of values of decisions, external decisions, and outcomes, respectively. For the sake of brevity we will assume further on that the external decisions z are given and represented as parameters of the mapping F.

A reader familiar with mathematical programming may be surprised, that such a model does not contain any goal function. This is done on purpose, and it is the recommended way of implementation of any model-based DSS. The core model (often called also substantive model) should include only logical and physical relations that are necessary to adequately represent relations between inputs x and outputs y. In addition to inputs and outputs, a model contains various intermediate and parametric variables (balance and/or state variables, resources, external decisions), conventionally called auxiliary variables. In a typical complex model decision and output variables are a small fraction of all variables. However, auxiliary variables are introduced for easing the model specification and handling, and are typically not interesting for an end-user of the model.

In other words, the core model is composed of decision, output and auxiliary variables, and of constraining relations (inequalities, equations, etc.) between these variables that indirectly determine the set of admissible (feasible) decisions and corresponding solutions. Some of the constraints may reflect the logic of handling events represented by variables. For example, in the RWQM model there are constraints:

$$\sum_{k \in K(j)} x_{jk} = 1 \qquad x_{jk} \in \{0, 1\}, \qquad j \in E \tag{1}$$

where $K(j)$ is the set of technologies considered for emission node j, and E is the set of nodes where emissions occur. This condition assures that exactly one technology is selected in each water treatment plant.

Therefore, the model defines a set of feasible decisions $X_0 \subseteq E_x$. In other words, x is feasible, if and only if $x \in X_0$. The set X_0 is usually defined only implicitly by a specification of a set of constraints that correspond to logical and physical relations among all the variables used in the model.

We shall explain now, why the model should not contain any representation of a preferential structure of a DM.

It is usually not possible to specify uniquely a model that can yield a unique solution reflecting the preferences of a DM. For example, very often it is practically impossible (even for a good analyst or an experienced DM) to specify e.g. values for a group of constraints that would cause a feasible solution that corresponds well to preferences of a DM. In order to illustrate this point let us consider the RWQM model. A DM typically considers different waste water treatment technologies and the related costs, as well as standards for water quality. However he/she knows that specification of constraints for a group of (either ambient or effluent) water standards may lead to solutions that are too expensive. On the other hand, assuming constraints for costs (with water quality standards being goals) could result in an unacceptable water quality. Values of constraints are in such cases formally parameters in a corresponding optimization problem. But those values are in fact decisions that reflect the preference structure of a user. Setting constraints' value too tight would result in restricting the analysis of the problem to a (possibly small) part of feasible solutions (often making the set X_0 empty). A typical advice in such situations is to specify two types of constraints, so called hard and soft constraints which correspond to *must* and *should* types of conditions, respectively. But, in fact, dealing with soft constraints can easily be done within multiobjective model analysis, which will be discussed later.

Therefore, the specification of a model that defines X_0 should not include any relations that reflect conditions for acceptability of a solution by a user or a preferential structure of a DM. In the RWQM, both costs and water quality standards are treated as objectives (criteria). This provides flexibility of examining trade-offs between costs and water quality. Hence, the so-called *core model* accounts only for logical and physical relations between all the variables that define the set X_0 of feasible solutions. The specification and parameters of the core model are not to be changed after a verification and validation of the model is done. All other constraints and conditions that implicitly define acceptability of a solution by a user and those that represent a preferential structure of a DM will be included into an interactive procedure of the model analysis.

Such an approach to model specification and analysis allows to design and implement a model-based DSS, which is conceptually composed of two parts:

- A constant and usually large core model. This part is built and verified before an actual analysis of a problem starts.
- A part that corresponds to a current specification of preferences defined by a user. This specification is interactively being changed, often drastically, by a DM.

Proper implementation of such an approach makes it possible for a DM to analyze feasible solutions that correspond to a his/her preference structure. Changing this structure is the essence of the model analysis and of the model-based decision support. An additional bonus is due to the fact that there always exists a feasible solution of the underlying mathematical programming problem, which is a prerequisite for an analysis of complex models.

Finally, we should point out that the value of a mathematical model as a decision aid comes from its ability to adequately represent reality. Therefore, there is always a trade-off between the requested accuracy (realism) of the model and the costs (also time) of its development and providing the model with data. Hence the requested accuracy should be consistent with the accuracy really needed for the model and with the quality of the available data. These issue is discussed in more detail in article 4.20.4.2.

2.3 Sensitivity analysis

In mathematical programming sensitivity analysis is typically understood as analysis of changes of an optimal solution caused by change of the data in the model. A traditional approach of such analysis is based on properties of an optimal solution. It typically consists of calculations of ranges of changes of parameters for which an optimal solution does not change, and on using a dual solution for calculations of changes of value of a goal function for changes of some parameters that are small enough for allowing such a simple evaluation procedure. These methodological topics, which all form the subject of post-optimal analysis, and the corresponding software tools have been extensively developed, especially for LP type of problems. However, their applicability is practically limited to rather small, linear models. Moreover, there are several problems with applications of these approaches to real problems. We restrict the discussion to the three basic issues:

- The concept and tools for sensitivity analysis have been developed and implemented for analysis of rather small models. However, models are larger and larger, therefore use of these technique is either cumbersome or practically impossible for many models.
- The range of changes of parameters for which the classical sensitivity analysis is valid for mixed-integer and non-linear types of model is typically too small to justify its application.
- In many models the quality of dual solution is rather questionable, for many other models the dual solution is practically non-unique. It is due to the fact that most of large models are numerically badly conditioned, and due to efficient presolve algorithms, which greatly decrease the resources (time and memory) needed for solving large problems. However they always guarantee the quality of the primal solution but often result in unreliable dual solution, which is the basis for classical sensitivity analysis. Therefore, such an analysis requires a good understanding of various techniques and corresponding tools, which is rather limited to highly skilled specialists in mathematical programming.

Given the above summarized reservations and limitations of sensitivity analysis in mathematical programming one should stress that the problem is much broader and complex in multiobjective decision support. An additional dimension of the problem is caused by the fact, that the nature of such a support requires solutions of many more parametric optimization problems than is needed by more restricted single-criterion optimization approaches. However, multiobjective decision support offers better ways for providing some of the functionalities that are theoretically promised by sensitivity analysis.

Here one should distinguish two groups of problems which are related to the two related but distinct issues, namely:

- Model development, where some parameters of the model can hardly be precisely determined; here by parameters we understand only coefficients in logical and physical relations;
- Model analysis, where a classical single-objective optimization-based approach forces to treat all but one actual criteria as constraints;

A discussion on how and when a selection of a type of model (such as fuzzy or stochastic) can adequately represent a problem, for which a deterministic model with fixed parameters may be too simplified is far beyond the scope of this article. Here we briefly outline how an appropriate model specification can help in sensitivity analysis during the multicriteria model analysis. In many practical applications a deterministic model is an adequate simplification provided that the developers of the model have enough data and experience to properly evaluate values of parameters. In some situations a parametric analysis of a model is nevertheless needed, but this is typically done during the model validation. Another technique which is useful, and is more efficient than some elements of sensitivity analysis, is a specification of so-called *soft constraints*, and to use such constraints for a definition of some output variables.

The the second issue is solved by the nature of the multiobjective model analysis, which is based on a core model that does not include preferential model of a user. As illustrated by the earlier discussion on the RWQM model, in classical single-criterion optimization typically several objectives were treated as constraints, for which one had to specify an acceptable value. This approach has not only the disadvantages discussed earlier, but it also requires analysis of the impact of changes caused by specified constraining values for criteria that are treated as constraints. Such values cannot be in practice specified precisely, therefore their modifications are inevitable. Sensitivity analysis was developed in order to help in analysis of such modifications. However, the functionality of sensitivity analysis, which was applied to this part of classical analysis of optimization models is replaced in multiobjective model analysis by more robust and natural approaches to problem analysis.

3 Multiobjective Model Analysis

3.1 Classical methods of model analysis

Multiobjective decision support builds on the classical methods of model analysis, therefore it can hardly be explained without an overview of these methods.

There is a general agreement on that there are generally two classical approaches to a model analysis: descriptive (sometimes called predictive) and prescriptive (normative). The descriptive models are used for prediction of the modeled system behavior without an attempt to influence it. The prescriptive models are aimed at providing information about controls (in managerial situations called decisions) which can result in a desired behavior of the modeled system. In other words, a descriptive DSS helps to answer a question such as *"what will happen if"* whereas a prescriptive DSS supports answers for questions like *"what decisions are likely to be the best"*. For the prescriptive type of analysis, optimization techniques were widely considered (especially in the OR community) as good tools for selecting (out of an admissible set) a solution (part of which is composed of the above mentioned controls) which is considered *"the best"*. The term *"best"* corresponds to a solution that provides best value of a performance index (goal function, objective, criterion) or a set of such indices that are used for evaluation of the expected consequences of implementing corresponding decisions (controls). A commonly accepted (by OR community) approach used to assume that a mathematical model corresponds well to reality and it is possible to define a performance index that reflects well the preferences of a DM; hence, an application of an optimal solution for the mathematical model could cause in reality *"the best"* results. However, many applications of such an approach to more complex problems have demonstrated that such an assumption is usually erroneous and optimization alone usually does not provide *"the best"* solution.

Descriptive methods of model analysis are very useful for model verification. The model should not only conform to the formal specification but also all discrepancies between intuitive judgment of a DM and analytic results obtained from the model must be resolved. In other words, such inconsistencies show that either the model (assumptions, specification, data) or the DM's intuition is wrong. Conflicts between results provided by the model and what is perceived by a DM must be resolved before the DM may trust the model, which is obviously a necessary (but often forgotten) condition for the actual use of a DSS.

Simulation is one of the tools useful for running a model in a descriptive mode. For simulation, one may use random values for variables or assign values basing either on DM's intuition or on a heuristic (possibly based on information from a knowledge base). So-called inverse simulation is a very useful technique for examination of decisions in situations, when specification of a set of feasible decisions is not easy. Obviously, a mixture of these techniques can be used for two groups of variables since values of the variables selected (into the group of simulated variables) can be temporary fixed. Various simulation techniques applied in the descriptive mode may provide information not only for model verification but may also lead the DM to modification of selected constraints or goals.

The usage of simulation and optimization can be compared as follows:

- In simulation mode decision variables are inputs and goals are outcomes. Therefore this technique is good for exploring intuition of a DM, not only for verification of the model but also for providing a DM with information about consequences of applying certain decisions (for example, what would be the value of goals and constraints). One can also consider simulation as an alternative-focused method of analysis that is oriented to identify (examine) the alternatives.
- Optimization can be considered as a goal-oriented (value-focused) approach that is directed towards creating alternatives. Optimization is driven by hope to reach a set of goals (however, not in the sense of goal programming). Therefore goals are a driving force and the values of decision variables are outcomes. This is very appealing. However, one should not forget about disadvantages that are consequences of the two facts. First, usually not all real objectives are included into the goals formulated for the optimization. Second, an optimal solution is found on a set of feasible solutions and a DM might prefer to change (usually implicitly) this set, if he knew that, for example, a substantial increase of a goal considered to be of secondary importance may result from a relatively small change of one of goals that is represented in the model as a constraint.

Therefore, interchangeable use of both (simulation and optimization) techniques has obvious advantages, but implementation of both techniques in classical models was difficult. This has resulted in the development of multiobjective model analysis methods and tools that combine advantages of these two classical approaches.

3.2 Aspiration-based analysis of efficient solutions

Many commonly known shortcomings of single-objective optimization and limitations of simulation based approaches have led to the development of various methodologies and techniques for multi-objective model analysis for decision support. We will briefly summarize here only one of these techniques, which seems to be the most natural method that best corresponds to a real-life DMP. This is the ARBDS (Aspiration-Reservation Based Decision Support) method which is an extension of the *aspiration level* (sometimes referred to as *reference point*) approach, and includes several other extensions or similar approaches that have been proposed and implemented in the last about 30 years.

A critical step in multi-objective analysis is the generation of part of the Pareto-optimal solution set that is interesting for the user. Efficient or Pareto-optimal solutions are those where an improvement in the value of one criterion cannot be attained without worsening the value of at least one other criterion. Generating the entire Pareto set is practically impossible. Therefore, most multi-objective methods facilitate the generation of some selected set of Pareto solutions – in our case, implicitly defined by aspirations and reservations specified by the user. Additional tools for analyzing these solutions and for generating other subsets of Pareto-optimal solutions based on these results are also needed. Since aspirations are usually not attainable, the decision maker must learn, while using the mechanisms of the method, how to adjust them in order to find a feasible solution that best meets her/his expectations.

The reference point method is based on the concept of satisficing behavior (also called bounded rationality), in which the decision maker attempts to attain aspiration levels, usually by first trying to improve the criterion that shows the worst performance. The reference point methods could be described by the following steps:

1. The user or decision maker (DM) selects out of outcome variables a number of criteria (objectives). In typical applications there are 2–7 criteria that are analyzed at a time.
2. The DM specifies an aspiration vector $\bar{\mathbf{q}} = \{\bar{q}_1, \dots, \bar{q}_k\}$, where $\bar{q}_i$ are aspiration levels (the desired values for each criterion) and k is the number of criteria. Additionally, the DM specifies a reservation vector $\bar{\bar{\mathbf{q}}}$, which is composed of the worst values of criteria that a DM would like to consider. Optionally, the DM can specify his/her preferences for criteria values between aspiration and reservation levels. This is done by interactive definition of levels of satisfaction for selected values of the corresponding criterion.
3. The underlying formulation of the problem is the maximization of a (piece-wise linear) achievement function. This can be interpreted either as a value function of the DSS specified in response to the specific aspiration and reservation levels, or as an ad-hoc, nonstationary approximation of the value function of the decision maker, dependent on these levels. The problem is then transformed by the DSS into an auxiliary parametric single-objective problem, the solution of which gives a Pareto-optimal point.[2]
4. The DM explores various Pareto-optimal solutions by changing selected elements of the aspiration $\bar{\mathbf{q}}$ and reservation $\bar{\bar{\mathbf{q}}}$ vectors that correspond to modifications of his/her preferences reflecting various trade-offs between attainable values of conflicting criteria. Additionally, a DM may stabilize a criterion (i.e., specify a desired value instead of minimizing or maximizing the value of this criterion) or temporarily remove a criterion from the analysis. This results in the computation of a Pareto optimal point with respect to the remaining "active" criteria, but values of inactive criteria are still available for review.
5. The procedure described in points 2, 3, and 4 is repeated until a set of satisfactory solutions is found.

The reference point approach may be considered as an extension of goal programming

[2]We refer to properly Pareto-optimal solutions with a prior bound on trade-off coefficients as Pareto solutions (unless otherwise mentioned). A Pareto-optimal point in objective space is composed of values of all criteria for a corresponding Pareto-optimal solution. If a specified aspiration level $\bar{q}$ is not attainable, then the Pareto-optimal point is the nearest (in the sense of a Chebyshev weighted norm) to the aspiration level. If the aspiration level is attainable, then the Pareto-optimal point is uniformly better than $\bar{q}$. Properties of the Pareto-optimal point depend on the localization of the reference point (aspiration and reservation levels) associated with the criteria.

which was the precursor of most multi-criteria optimization and model analysis methods. One of important advantages of ARBDS methods is their ability to always provide an efficient solution (goal programming methods, which minimize the distance from a given goal, provide non-efficient (dominated) solution when given goals are attainable). Another advantage of ARBDS is due to the natural way of defining implicitly by a user the weighting coefficients in the weighted Chebyshev norm (by a specification of both aspiration and reservation values) whereas the goal programming methods have to rely on either automatic or built-in methods of scaling.

The reference point approach to multi-objective optimization can also be used for inverse simulation: instead of repeatedly adjusting the decision variables in order to determine acceptable states (expressed as constraints in the classical approach to optimization), the user chooses desired states (in terms of ranges of values of objectives) and the DSS determines for her/him the resulting values of the decision variables. The reference point approach also takes into account soft constraints often needed in the single-criterion optimization. Namely, one can replace a soft constraint (or group of constraints) by an objective, and then set the aspiration level equal to the desired value of the constraint and the reservation level to the worst acceptable value. Thus, violations of soft constraints can be treated as criteria (to be minimized) in the multi-objective approach.

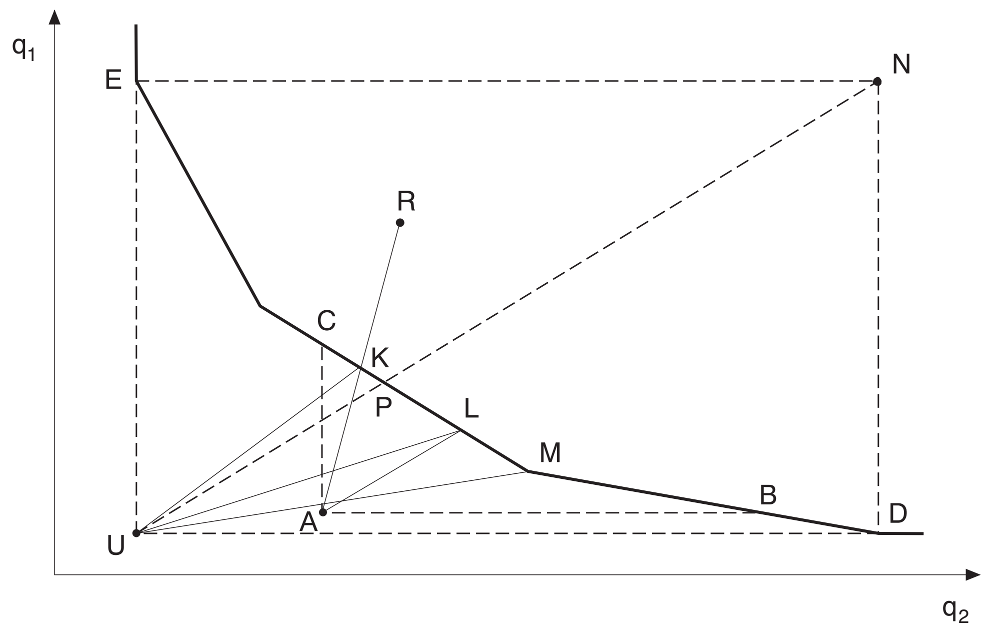

Figure 2: Illustration of a Pareto-optimal surface, and of various selections of efficient solutions for a two criteria case.

Geometrical aspects of the reference point approach are shown in Figure 2, which illustrates a Pareto-optimal frontier (between points D and E) for a two-criteria minimization problem. Utopia and nadir points (denoted by U and N, respectively) are composed of the best and worst values of criteria in the Pareto frontier.

Typically the analysis starts with so-called the neutral compromise solution, denoted by the point P, which is computed automatically by using aspirations and reservations set to the corresponding utopia and nadir values.

For a given aspiration point A, various Pareto-optimal points are obtained as closest to the non-attainable aspiration level A, depending on the definition of achievement scalarizing function, which is used by the underlying procedure of converting multicriteria optimization problem into a parametric single-criterion problem whose solution provides a Pareto-optimal solution that corresponds best to the current specification of DM's preferences.

For example, the achievement scalarizing function can take the form:

$$\sigma(\mathbf{q}, \bar{\mathbf{q}}, \bar{\bar{\mathbf{q}}}) = \min_{1 \le i \le k} \sigma_i(q_i, \bar{q}_i, \bar{\bar{q}}_i) + \epsilon \sum_{i=1}^{k} \sigma_i(q_i, \bar{q}_i, \bar{\bar{q}}_i), \tag{2}$$

where $q, \bar{q}, \bar{\bar{q}}$ are vectors of values of criteria, aspiration levels, and reservation levels, respectively, and ϵ is a given small positive number. Maximization of the function (2) provides a properly Pareto-optimal solution with the trade-off coefficient smaller than $(1 + 1/\epsilon)$. Partial or component achievement functions $\sigma_i(\cdot)$ are strictly concave functions of the criteria vector components q_i and can have the same analytical form for the three types of criteria, i.e., minimized, maximized, and goal. Namely, for the MCMA software illustrated in Figure 3, PWL functions are used. Such functions are defined by utopia and nadir points (computed automatically) and by user preferences specified through a graphical user interface. These preferences include at least a specification of aspiration and reservation values for each criterion. Optionally, as discussed later in this article, preferences for additionally selected values of criteria can be specified.

Here we only briefly outline the basic properties of the component achievement functions, which are necessary to understand the background of the multi-objective decision support. Such functions are strictly monotone (decreasing for minimized and increasing for maximized criteria, respectively) for the two most commonly used types of criteria, i.e. minimized and maximized. For the third type of criteria, i.e. goal type, a target value is specified and the corresponding component achievement function is strictly increasing for the criterion values smaller than the target value, and strictly decreasing otherwise. Moreover, for all types of criteria, the functions $\sigma_i(\cdot)$ have the following values for utopia, aspiration, reservation, and nadir points, respectively:

$$\sigma_i(q_i^U, \cdot) = 1 + \bar{\beta}, \quad \sigma_i(\bar{q}_i, \cdot) = 1, \quad \sigma_i(\bar{\bar{q}}_i, \cdot) = 0, \quad \sigma_i(q_i^N, \cdot) = -\bar{\eta} \tag{3}$$

where $\bar{\beta}$ and $\bar{\eta}$ are given positive parameters, typically equal to 0.1 and 10, respectively. However, in order to correctly handle aspiration and reservation values close to utopia and nadir values, respectively, these parameters should be dynamically adjusted.

As shown in Figure 2, using the achievement scalarizing function (2) and selecting the points A and R for the aspiration and reservation levels, respectively, would result in a Pareto-optimal solution denoted by K. With the aspiration level A, the application of different scalarizing functions (e.g., specified by using different reservation points) can result in any Pareto solution between the points B and C. For example, a scalarizing function that uses weighting coefficients corresponding to the difference between the utopia and nadir points would result in the point L, while another scalarizing function using only the utopia and the aspiration point for the definition of weighting coefficients would result in the point M. However, using the utopia and the nadir point for the automatic calculation of weighting coefficients has a number of drawbacks (especially for large problems, and for problem for which the values of utopia and nadir differ by several orders of magnitude), hence it is much better to implement an approach based on aspiration and reservation levels, which provides the user with the most natural way of examination of these parts of the Pareto-set that best corresponds to his/her preferences.

Finally, one should notice, that the described method works also for cases when aspiration and reservation levels are both either attainable or not attainable. In both cases the selected Pareto-optimal solution is defined in the same way as described above, i.e. by the intersection of the line defined by the aspiration and reservation points with the surface defined by all Pareto-optimal solutions. Of course, for the case when both aspiration and reservation points are attainable, such Pareto-optimal solution is uniformly better than the aspiration level, which is one of the important advantages of the described method over the classical goal programming approach (which provides as the solution the point that minimizes the distance from the aspiration point; therefore for attainable aspirations such solution is not Pareto-optimal).

3.3 Interactive multi-objective model analysis

While understanding of a background of the ARBDS methodology is easy and can be useful, its practical use requires an easy to be used interface for a specification of aspiration and reservation levels. Actually, such an interface may support also an optional specification of preferences in a more precise way, namely by a specification of a PWL component achievement functions.

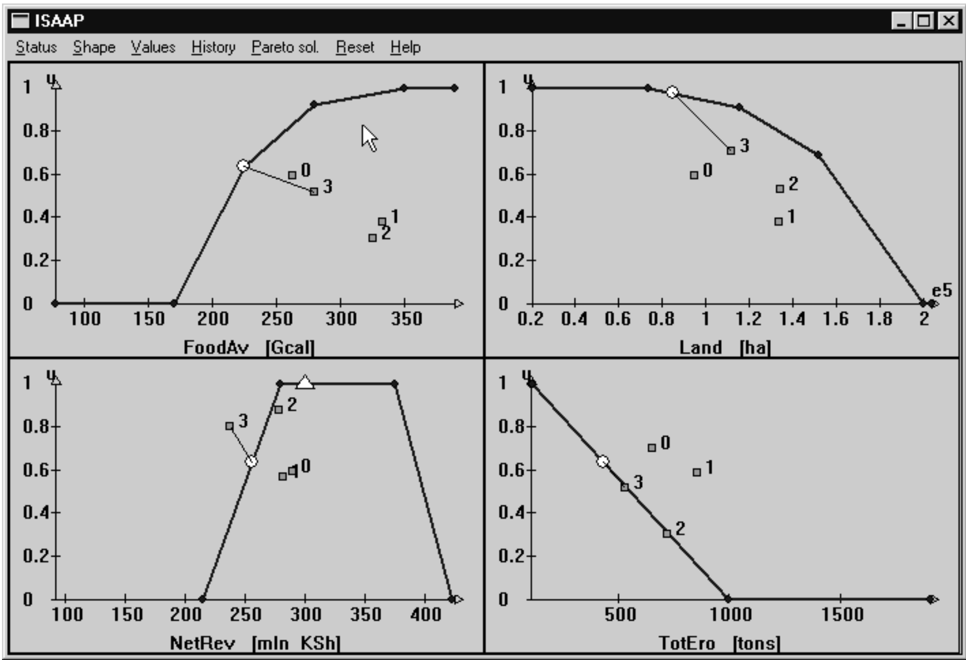

Figure 3: A user interface for graphical specification of user preferences.

A screen dump of such an interface is shown in Figure 3. In this example, which is taken from the Land Use Planning model discussed in article 4.20.4.2, four criteria are considered:

- FoodAv (average food production), which is maximized,
- Land (land usage), which is minimized,
- NetRev (net revenue), which is maximized,
- TotEro (total erosion), which is minimized,

In Figure 3 results of the fifth iteration are shown in the criteria space. The values of criteria for previous solutions are marked by squares numbered 0 (which denotes the compromise solution) through 4, and the values of criteria for the last solution are denoted by unmarked circles connected by a segment with the corresponding value of the previous solution. For the TotEro criterion the component achievement function is defined by aspiration and reservation levels (0 and 1000, respectively) only. The NetRev criterion has been stabilized (its target value is market by the white triangle). For other two criteria the user has defined more detailed specification of preferences. Namely, in addition to aspiration and reservation levels, the levels of satisfaction for additionally selected two values for each of these two criteria have been interactively defined (e.g. for the criterion FoodAv for criteria values about 220 and 270, to be about 0.6 and 0.9, respectively). In fact, more important than the values of the satisfaction levels is the shape of the PWL function which is defined by the selected points between the aspiration and reservation levels. Such function can be interpreted as a membership function used in the Fuzzy Sets method, which has an intuitive interpretation: its value denotes the degree of membership of an element to a set. Hence, if one defines a set of good/satisfactory solutions, then the elements having value of the membership function equal to 1 belong to the set, those with value equal to 0 do not belong to this set (i.e. are not good solutions) and elements with values of membership function between 0 and 1 are considered to be satisfactory to a degree represented by such a value.

The software for interactive multi-criteria model analysis supports of course many functions, which are needed for a multi-objective model analysis (such as selection of outcome variables and definition of criteria, support for analysis of history of solutions, changing status of criteria, etc.). However, due to a limited space these functions cannot be presented here.

3.4 Analysis of discrete alternatives

Analysis of a small set of discrete solutions under a finite number of conflicting criteria, is a specific type of decision problem. In some situations there are indeed only several alternatives out of which a DM has to select only one solution (for example, problem of a facility location, selection of a technology or of a car). However, in other situations a complex problem is analyzed by specialists who define several alternative solutions, which are presented to a DM. The latter approach is of course not recommended because it substantially limits the sovereignty of actual DM: even best specialists have different preferential structure than a DM, therefore the preselection of alternative that are presented to the DM can hardly cover all solutions that the DM should analyze.

One should point out, that the analysis of discrete alternatives can be done with the ARBDS method outlined above. For this purpose one need to define a simple model using binary variables corresponding to each alternative, and defining appropriate outcome variables. Such model is of MIP type, and its multicriteria analysis can be done using the methodology and tools described in this article.

However, there are several methods specialized for analysis of alternatives, each of them preferred by some scientists and practitioners, and vigorously criticized by those scientists and practitioners, who strongly believe in advantages of alternative approaches. We outline here only two of such specialized methods, however without comments on their advantages and limitations:

1. The Simple Multi-Attribute Rating Technique (SMART). The performance of each alternative is expressed in grades on a numerical scale, which are evaluated through a direct-rating procedure. In practical applications the grades are typically interpreted as category labels (such as cheap, somewhat more expensive, more expensive, etc, or excellent, good, fair, unsatisfactory).
2. The Analytic Hierarchy Process (AHP). The alternatives are examined in pairs. Their

relative performance is evaluated as a ratio of subjective values (Multiplicative AHP) or as a difference of grades (Additive AHP). The Additive AHP can be considered as SMART with pairwise comparisons.

More complex approaches are applied for supporting group decision making, where in addition to the complexity of multicriteria problem analysis by a single DM, one has to consider various problems typically for decision making by a heterogeneous group of DM. This topic is far beyond the scope of this article, hence we only mention here a selection of types of problems that should be taken into account: negotiations procedures, forming coalitions, the anatomy of power, scale values for relative power, weighted judgments, searching a compromise solutions.

4 Structure and Use of Model-Based DSSs

An implementation of any DSS is always problem specific. However, a general structure of model-based decision support systems can be illustrated in Figure 4.

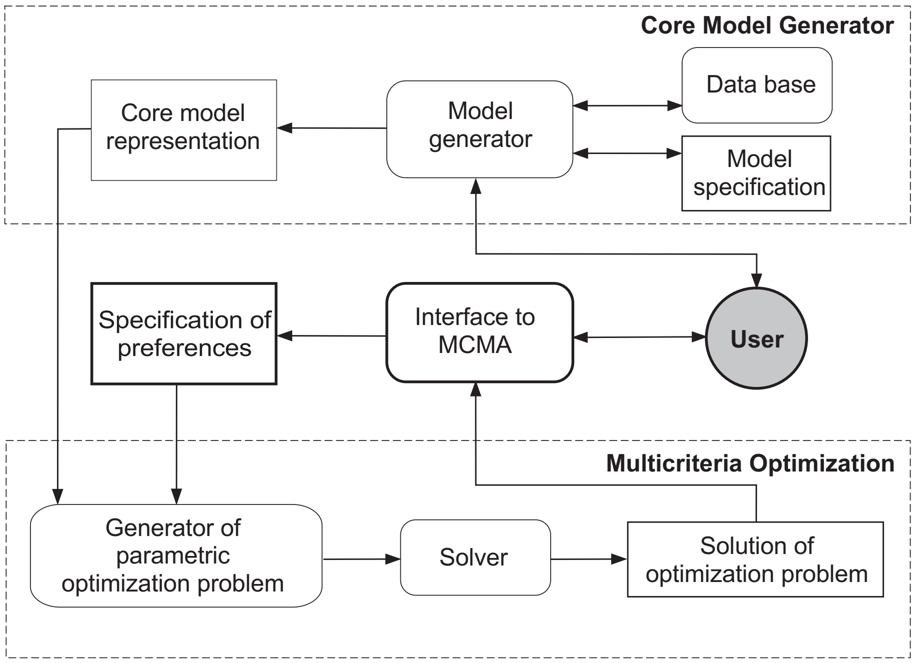

Figure 4: General structure of model-based decision support.

A model-based DSS is composed of three functional parts that are mutually linked:

- The main (from the point of view of the user) component is an interface to MCMA, which helps to analyze solutions and to modify user preferences. The solutions are typically represented by values of selected outcome variables but additionally more detailed analysis of full solutions should be available for a more detailed, and typically problem specific, examination. For the ARBDS approach the preferences are represented by the definitions of criteria, and the corresponding aspiration, and reservation levels. The main part of the problem analysis consists of many iterations described in Section 3.2, which help the user to learn about the range of attainable goals (aspirations) and examine trade-offs between values of conflicting objectives. Such analysis is often augmented by a simulation-based scenario analysis. Moreover, soft simulation can be included in

MCMA (by defining a criterion which corresponds to a distance between given and actual values of decision variables).

- Model generator is developed in the preparatory stage of the development of a DSS. This stage is of a more technical nature and for complex problems it is typically done by analysts. However, the user has to be involved in the core model specification process in order to make sure that the assumptions that have to be made for any model specification correspond well to her/his understanding of the model representation of the decision problem. It is important to stress again that the core model includes only physical and logical relations, and should not include constraints that represent the preferential structure of the DM.
- Multicriteria optimization module is of a purely technical nature and it is typically hidden from the user. Its role is to generate a parametric single criterion optimization problem based on the core (substantive) model and on the aspiration and reservation levels (or on a more precise specification of preferences) that represent the current preferential structure of a DM. Such an optimization problem is then solved by a solver, which typically is a general purpose software specialized for a corresponding type of optimization problem (e.g. LP, MIP or NLP). The optimal solution of such an auxiliary optimization problem provides a Pareto-optimal solution that has properties implicitly defined by the preferences specified by the user. One should note that for every properly specified core model (which represents only physical and logical relations) there is always a feasible solution of the auxiliary optimization problem, even if the preferences of the user are not attainable.

All elements of the software illustrated in Figure 4 are mutually linked by a data interchange tool, which is hidden from the user. Such a tool should provide an easy and efficient way for defining and modifying optimization problems, as well as interchanging of data between a problem generator, a solver, and software modules that serve for problem modification and solution analysis. Therefore the user can concentrate on the most important feature of DSSs, namely on analysis of consequences of implementation of decisions that correspond to his/her preferences that he/she changes during the analysis of the decision problem.

5 Advantages and Limitations of Model-Based DSS

The key to a discussion of the role of any successful decision support method is composed of understanding the following crucial elements of actual decision-making:

- Whenever a complex problem is considered, then no one individual system of management is sufficient, it is the integration of their ideas that creates a powerful force for change and improvement.
- Experienced DMs have a fairly stable set of ways for thinking, evaluating, judging and making decisions; this is often called a Habitual Domain (HD). HD is specific for each person, and it differs often substantially, even for persons with similar cultural and professional background, experience and duties.
- DMs typically need help only for a part of the DMP, where intuition and experience are not enough for identifying decisions that best correspond to goals and preferences of the DM. In particular, help is needed for learning about attainable goals and decisions that lead to them, for examination of consequences of given decisions, and for analysis of trade-offs between conflicting objectives.

- The decisions are made by a DM, who takes responsibility for consequences of their implementation. Therefore any DSS has to respect the sovereignty of the DM.

The OR methods were developed and first applied to well structured decision problems, where optimization focused approaches provided desired solutions. However, already about 50 years ago H. Simon has pointed out that actual decisions, particularly in large organizations, are optimized only until a satisficing level of key criteria is reached; at this point instead of a further optimization other attributes of decisions are considered. Due to the successes of early applications and limitations of computing power, OR for relatively long time was focused on single-criterion optimization approaches, which has limited application to actual decision-making support for complex problems. The development various methods of multi-objective model analysis was stimulated by the demand for decision support in complex problems.

Because the variety of decision problems and of habitual domains of DMs, there will probably never be just one method of model-based decision support. In fact, no single modeling paradigm alone is sufficiently good enough to identify and analyze various enough policy options for any complex decision problems that are necessary for making rational decisions. Rather, an integration of various modeling methods and tools is needed to provide the best available support possible to analyze complex problems.

Lessons learned from the applications of various modeling paradigms to very different types of real-world problems, and the recent abundance of computing hardware and software tools makes it possible to integrate several methods of a model specification and analysis, and to apply them to large and complex problems. Such an integration calls for a collaboration of specialists, who have been concentrated – and therefore have substantial experience – in a particular method. Therefore, one should expect that various integrations of different modeling paradigms will be used more broadly to improve decision-making support in a wide range of practical problems. However, the key role in actual decision making will stay with human decision makers.

6 Bibliography

Below is a selection of 10 key bibliography items. The bibliography has been generated from the BibTEX database using the ACM citation and bibliography styles. It can be easily generated in one of many commonly used styles, if this would help processing this article.

Here are the annotations to each position:

(1): This book presents a practical introduction and guide to use of Operations Research techniques in scientific decision-making, design and management.

(2): This book presents a collection of examples and arguments on uniqueness of human beings and limitations of using computers.

(3): This is a collection of articles providing state of the art reviews and the most recent advances of fundamental theories, methodologies and applications of multicriteria decision support.

(4): This is a collection of articles on multi objective linear programming and interactive methods, as well as on preferences and learning during decision analysis; it also contains several articles on applications.

(5): This book presents methods and examples of multi-criteria analysis of a set of alternatives by applying for preferential judgment estimations of ratios of subjective values.

(6): This is collection of articles providing conceptual and operational understanding of the nature of models as representations of reality and as tools for description and analysis of decision-making problems in various areas.

(7): The classical book describing how people make actual decisions, particularly in large organizations; the author developed the concept of satisficing decisions and shown that actual decision makers, through learning, adaptively develop aspiration levels for various criteria.

(8): This volume contains a collection of articles dealing with various methodologies of multi-criteria decision support, including preference modeling, negotiation and group decision support, system and philosophical issues, as well as descriptions of applications in environmental and natural resources management problems.

(9): The monograph introduces the methodological background and describes various features of the decision environment and the ways in which model-based decision support can help the modern decision making process; it presents the methodology and software tools for building mathematical models and for their multicriteria analysis; the presented methods and tools are illustrated by detailed presentation of four complex environmental applications.

(10): The book presents all aspects of habitual domains: their foundations, expansion, dynamics and applications to various important problems in people's lives, including effective decision making. Based on an integration of psychology, system science, management and common sense and wisdom, the book provides a simple but unified set of tools to understand the human behavior mechanism.

References

[1] Carter, M., and Price, C. *Operations Research.* CRC Press, New York, 2001.

[2] Dreyfus, H. *What Computers Still Can't Do. A Critique of Artificial Reason.* The MIT Press, Cambridge, 1999.

[3] Gal, T., stewart, T., and Hanne, T., Eds. *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications.* Kluwer Academic Publishers, Boston, London, 1999.

[4] Karwan, M., Sprong, J., and Wallenius, J., Eds. *Essays in Decision Making.* Springer, Berlin, New York, 1997.

[5] Lootsma, F., Ed. *Multi Criteria Decision Analysis via Ratio and Difference Judgement*, vol. 29 of *Applied Optimization.* Kluwer Academic Publishers, Boston, London, 1999.

[6] Richardson, J., Ed. *Models of Reality.* Lomond Books, Mt. Airy, 1987.

[7] Simon, H. *Models of Man.* J. Wiley & Sons, Chichester, New York, 1957.

[8] Stewart, T., and van den Honert, R., Eds. *Trends in Multiple Criteria Decision Making*, vol. 465 of *Lecture Notes in Economics and Mathematical Systems.* Springer Verlag, Berlin, New York, 1998.

[9] Wierzbicki, A., Makowski, M., and Wessels, J., Eds. *Model-Based Decision Support Methodology with Environmental Applications*. Series: Mathematical Modeling and Applications. Kluwer Academic Publishers, Dordrecht, 2000. ISBN 0-7923-6327-2.

[10] Yu, P. *Forming Winning Strategies, An Integrated Theory of Habitual Domains*. Springer Verlag, Berlin, New York, 1990.