

Interactive Specification and Analysis of Aspiration-Based Preferences

Janusz Granat

*Institute of Control and Computation Engineering, Warsaw University of
Technology, Warsaw, Poland and Institute of Telecommunications,
Warsaw, Poland*

Marek Makowski

International Institute for Applied Systems Analysis, Laxenburg, Austria

RR-00-09

March 2000

Reprinted from *European Journal of Operational Research* 122 (2000)
469–485.

Research Reports, which record research conducted at IIASA, are independently reviewed before publication. Views or opinions expressed herein do not necessarily represent those of the Institute, its National Member Organizations, or other organizations supporting the work.

Reprinted from *European Journal of Operational Research* 122 (2000) 469–485.
Copyright © (2000), with permission from Elsevier Science.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage or retrieval system, without permission in writing from the copyright holder.

Interactive specification and analysis of aspiration-based preferences

Janusz Granat^{a,b,1}, Marek Makowski^{c,*}

^a *Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland*

^b *Institute of Telecommunications, Szachowa 1, Warsaw, Poland*

^c *International Institute for Applied Systems Analysis, Schlossplatz 1, A-2361 Laxenburg, Austria*

Received 1 October 1998; accepted 1 April 1999

Abstract

Model-based Decision Support Systems (DSSs) often use multi-criteria optimization for selecting Pareto-optimal solutions. Such a selection is based on the interactive specification of user preferences. This can be done by a specification of aspiration and reservation levels for criteria. Diverse Graphical User Interfaces (GUIs) can be used for specification of these levels, as well as for the interpretation of results. In the approach presented in this paper, the specified aspiration and reservation levels are used for the generation of component achievement functions for corresponding criteria, which reflect the degree of satisfaction with given values of criteria. This paper outlines the methodological background and modular structure of a tool ISAAP (which is included in MCMA) for multi-criteria analysis of decision problems that can be represented as Linear Programming (LP) or Mixed Integer Programming (MIP) problems. The MCMA has been used at IIASA for the analysis of decision problems in water quality management and land use for sustainable development planning. These experiences have shown that MCMA tool is applicable also to large LP and MIP problems. Other implementations of the same methodology have also been applied to analysis of non-linear problems in several engineering applications. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Decision Support Systems; Modeling; Multi-criteria analysis; Software

1. Analytical modeling in decision support

A Decision Support System (DSS) is a computerized tool which helps analyze a decision problem. Model-based decision support often uses

multi-criteria optimization for selecting Pareto-optimal solutions. Such a selection is based on the interactive specification of user preferences. This paper outlines the methodological background and presents a tool that provides a graphical interface for interactive multi-criteria model analysis. The presented tool has been applied to teaching and to several real-world applications which has proven its applicability to both small tutorial and large real-world models.

* Corresponding author. Tel.: +43-2236-8070; fax: +43-2236-71313; URL: <http://www.iiasa.ac.at/~marek>.

E-mail address: marek@iiasa.ac.at (M. Makowski).

¹ URL: <http://www.ia.pw.edu.pl/~janusz>

For any model-based DSS, one can distinguish the following two groups of related modeling activities, underlying methodologies and software:

Model generation: Generation of a *core model* (often referred to as a *substantive model*) which is a representation in terms of a mathematical model of all logical and physical relations between variables representing the decision problem being examined without specifying goal functions. The core model implicitly defines a set of feasible solutions but does not contain any preferential structure of the user. This structure is specified and later modified during the analysis of the model.

Model analysis: Selection from the set of all feasible solutions – implicitly defined by the *core model* – of a subset of acceptable solutions, and then a further selection from this subset of a solution that, hopefully, corresponds best to the preferences of the user. There is no easy and universal way to specify a representation of a preferential structure. This can be done by a selection of desired criteria values (in aspiration-led multi-criteria model analysis approaches); or by a selection of one criterion and imposing additional constraints for other criteria (in single-criterion optimization methods); or by a specification of desired values of decision variables (in simulation-based approaches). Each method of the representation of a preferential structure for a user has a number of parameters that have to be set by the user in order to formulate a corresponding optimization, or simulation, problem. The analysis is often done in an interactive way, thus allowing a user analysis of previously obtained solutions, changing the representation of his/her preferential structure, and thus formulating a corresponding underlying optimization problem. Model analysis requires the solution of a series of auxiliary optimization or simulation problems, which in turn requires robust and efficient solvers that can handle the related computational tasks in a way that is transparent for the user of a DSS.

We will concentrate on the phase of multi-criteria model analysis based on an extension of the aspiration-led multi-criteria optimization based model analysis, commonly called Aspiration-Reservation Based Decision Support (ARBDS). Today, ARBDS is one of the most promising

techniques for model analysis for decision support. However, one of the major constraints for the wide applications of any method that requires interaction with the user is the lack of modular software tools that can be used for an implementation of a problem-specific DSS. Therefore, ISAAP, the modular tool that facilitates the interaction with the user by providing all the functions necessary for interactive analysis of a problem using the ARBDS methodology, has been developed and is documented in this paper. The name ISAAP is an abbreviation of the full name Interactive Specification and Analysis of Aspiration-based Preferences, which characterizes the methodological background applied in the ISAAP.

2. Aspiration-Reservation Based Decision Support

Discussion of different approaches to decision-making support is clearly beyond the scope of this paper. The theoretical and methodological backgrounds for aspiration-based decision analysis and support, as well as a large bibliography on related topics, can be found e.g. in Wierzbicki et al. (2000). We will deal with one of the most successful – see e.g. Korhonen and Wallenius (1989) for a justification of this statement – class of DSS, namely with model-based DSS which use aspiration-led multi-objective optimization as a tool for computing and selecting efficient solutions. This approach, originally proposed in Wierzbicki (1980), now has more than a dozen slightly different methodological versions. A unified procedure that covers most of those approaches has been proposed by Gardiner and Steuer (1994).

An extension of the aspiration-led multi-criteria model analysis is called Aspiration-Reservation Based Decision Support (ARBDS). The ARBDS methodology has been implemented in a number of DSS presented in Lewandowski and Wierzbicki (1989). The relations between ARBDS and other approaches to multi-criteria optimization are discussed in more detail in Makowski (1994b). ARBDS can be also considered (see Ogryczak and Lahoda, 1992) as an extension of Goal Programming (see e.g. Charnes and Cooper (1967), for details) probably the oldest technique for multi-

criteria analysis of linear programs. Today, ARBDS is one of the most promising techniques for model-based decision support.

MCMA is the recent implementation of ARBDS methodology. The following two stages of the model analysis with the help of MCMA can be distinguished:

- First, a *core model* is specified and generated. The core model contains only a set of constraints that correspond to the logical and physical relationships between the variables used in the model. Those variables should also include ones that represent potential criteria (goals, performance indices). In the preparatory stage a user selects, from the core model variables, a set of criteria that will be used for the analysis of the model, and specifies a type for each criterion. The selected type declares that a criterion is either minimized, or maximized, or targeted at a given value (*goal-type* of a criterion). Note, that a variable can also represent more complicated forms of criteria such as following a trajectory, minimization of a distance, etc. Examples of different types of criteria, which are formally represented by a variable whose value is either minimized or maximized, and the way to handle so-called soft constraints in the framework of ARBDS can be found e.g. in Makowski (1994b). After the selection of a set of criteria, MCMA automatically performs a series of optimizations in order to compute the Utopia point and an approximation of the Nadir point. Utopia and Nadir points are vectors, in the space of criteria, composed of best and worst values of the criteria in the efficient set. Isermann and Steuer (1987) have shown that a computation of a Nadir point for problems with more than two criteria may be very difficult. In our approach, the Nadir point plays a minor informative role, it only bounds values of corresponding reservation levels; therefore, there is no justification for spending resources to compute a correct value of the Nadir. Hence, we assume as an approximation of Nadir, the worst value of a corresponding criterion obtained during the analysis. The preparatory stage is finished with computation of the so-called *compromise solution* which corresponds to a

problem for which the aspiration and reservation levels are automatically set to the Utopia and an approximation of the Nadir points, respectively.

- Second, an interactive procedure is used for helping the user to select various efficient solutions that best correspond to his/her preferences. During such procedure a user specifies goals and preferences represented by values of criteria that he/she wants to achieve and those that he/she wants to avoid. The vectors composed of those values are called *aspiration* and *reservation* levels, respectively. Such a specification defines component achievement functions (see Section 3) which are used for a selection of a Pareto-optimal solution. Such a solution is achieved by the generation of additional constraints and variables, which are added by MCMA to the *core model*, thus forming an optimization problem whose solution results in a Pareto-optimal solution that is nearest, in the sense of a measure defined by the aspiration and reservation levels, to the specified aspiration levels; if the aspiration levels are attainable, then a solution is uniformly better than these levels.

The ISAAP, which is a part of MCMA package, handles the interaction with the user in the second stage of the problem analysis; therefore, we will provide more details about this stage, which can be described in the form of the following steps:

1. The user specifies new aspiration and reservation levels for all active criteria. For each stabilized criterion (if any), the user specifies a corresponding target value, as well as aspiration and reservation levels for a deviation from the specified target value. Optionally, the user can specify his preferences for selected criteria by a piece-wise linear component achievement function.
2. The user can change the shape of the component achievement function corresponding to each criterion by either defining a piece-wise linear function for the criterion values between aspiration and reservation, or by stabilizing a criterion.
3. The user can change the status of each criterion. The default status can be changed to *inactive*, *disregarded* or back to the original status (which

is one of *min*, *max*, or *goal*, depending on the type of the criterion).

4. The user can analyze criteria values of the solutions computed so far, together with values of aspiration and reservation levels used for each solution. This part of the analysis is supported by the history options.
5. The user can store a currently analyzed solution of the underlying LP or MIP problem for a more detailed analysis, one that is typically problem specific.
6. The user can freely switch between the actions summarized above until he/she decides that his/her preferences are properly represented for the next optimization. Once the optimization is selected, the MCMA takes control of the program flow. MCMA generates a single-criterion optimization problem whose solution is a Pareto-optimal solution that corresponds to the current preference structure of the user and executes an appropriate solver, which computes such a solution. The user regains control of the program when the solution of the last specified problem is ready and added to the previously obtained solutions.

The steps described above are repeated in order to explore various Pareto-optimal solutions, until a satisfactory solution is found or until the user decides to break the analysis. In either case, the analysis can be continued at a later time from the last obtained solution.

3. Selection of a Pareto-optimal solution

Multi-criteria optimization methods typically assume that a multi-objective problem is converted into an auxiliary parametric single-objective problem whose solution provides a Pareto-optimal point. A solution is called Pareto-optimal, or efficient, solution if 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. In other words, one cannot improve any criterion without deteriorating a value of at least one other criterion.

The key problem here is obviously a selection of a particular Pareto-optimal solution out of typi-

cally large set of such solutions. This selection is implicitly determined by a conversion of a multi-objective problem into a parametric single-objective problem. Different multi-criteria optimization methods apply different conversions (see, e.g. Haimes and Hall, 1974; Wierzbicki, 1980; Sawaragi et al., 1985; Steuer, 1986) but most commonly known methods can be interpreted (see Makowski, 1994b) in the terms of the Achievement Scalarizing Function (ASF). The concept of ASF has been introduced by Wierzbicki (1977) and it is very useful for comparing different approaches to multi-criteria optimization.

The selection of a particular Pareto-optimal point is determined by the definition of the ASF, which – for the aspiration-led model analysis – also includes a selected aspiration point. Most of those methods use the maximization of an ASF in the form

$$s(q, \bar{q}, w) = \min_{1 \leq i \leq n} \{w_i(q_i - \bar{q}_i)\} + \epsilon \sum_{i=1}^n w_i(q_i - \bar{q}_i), \quad (1)$$

where $q(x) \in \mathbb{R}^n$ is a vector of criteria, $x \in X_0$ are variables defined by the core model, X_0 is set of feasible solutions implicitly defined by the core model, $\bar{q} \in \mathbb{R}^n$ is an aspiration point, $w_i > 0$ are scaling coefficients (see the comment below) and ϵ is a given small positive number. Maximization of (1) for $x \in X_0$ generates a properly efficient solution with the trade-off coefficients (as recomputed in terms of u_i defined below) smaller than $(1 + 1/\epsilon)$. For a non-attainable \bar{q} , the resulting Pareto-optimal solution is the nearest – in the sense of a Chebyshev weighted norm – to the specified aspiration level \bar{q} . If \bar{q} is attainable, then the Pareto-optimal solution is uniformly better. Setting a value of ϵ is itself a trade-off between getting an overly restricted set of properly Pareto-optimal solutions or an overly wide set that is practically equivalent to weakly Pareto-optimal solutions. Assuming the ϵ parameter to be of a technical nature, the selection of efficient solutions is controlled by the two vector parameters: \bar{q} and w .

There is a common agreement that the aspiration point is a very good controlling parameter for

examining a Pareto-optimal set. Much less attention is given to the problem of defining the scaling coefficients w . Note that the coefficients w should not be confused with the weights used by some methods for conversion of a multi-criteria problem into a single-criterion problem with a weighted sum of original criteria. In the function (1), coefficients w play a different role than in a weighted sum of criteria. A detailed discussion on weights in a scalarizing function is beyond the scope of this paper. The four commonly used approaches are summarized in Makowski (1994b). In practical applications, the most promising approach is based on the calculation of weights used in definition of the Chebyshev norm mentioned above, with the help of the aspiration level \bar{q} and a reservation level \underline{q} , the latter being composed of values of criteria that the user wants to avoid. This is the ARBDS approach that has been introduced by the DIDAS family of DSS described in Lewandowski and Wierzbicki (1989). More detailed arguments against approaches based on the weighted sum of criteria are provided by Makowski (1994b) and by Nakayama (1994).

The ASF for the ARBDS approach usually takes the form

$$\mathcal{S}(q, \bar{q}, \underline{q}) = \min_{1 \leq i \leq n} u_i(q_i, \bar{q}_i, \underline{q}_i) + \epsilon \sum_{i=1}^n u_i(q_i, \bar{q}_i, \underline{q}_i), \tag{2}$$

where \bar{q} and \underline{q} are vectors (composed of \bar{q}_i and \underline{q}_i , respectively) of aspiration and reservation levels, respectively, and $u_i(q_i, \bar{q}_i, \underline{q}_i)$ are the corresponding Component Achievement Functions CAF (defined later in detail), which can be simply interpreted as non-linear monotone transformations of q_i taking into account the information represented by \bar{q}_i and \underline{q}_i . Maximization of the function (2) over the set of feasible solutions X_0 defined by the corresponding core model provides a properly Pareto-optimal solution with the properties discussed above for the function (1).

The ASF implemented in MCMA is a modification of function (2). The modification has been stimulated by some applications for which it is often useful to temporarily disregard some of the criteria. A criterion for which the user does not

wish to define the corresponding component scalarizing function is called in MCMA an *inactive criterion*. Inactive criteria are also useful for computing a good approximation of a Nadir point. However, completely disregarding a criterion from the ASF may result in both numerical problems, caused by a degenerated problem, and in a random value of the criterion, which may be unnecessarily bad and can in turn result in a bad approximation of a Nadir point (see Makowski (1994b), for more details). Therefore, the following form of the ASF is implemented in MCMA in order to facilitate a proper handling of inactive criteria:

$$\begin{aligned} \mathcal{S}(q, \bar{q}, \underline{q}) = & \min_{i \in I} u_i(q_i, \bar{q}_i, \underline{q}_i) \\ & + \epsilon \sum_{i \in I} u_i(q_i, \bar{q}_i, \underline{q}_i) \\ & + \epsilon \sum_{i \in \bar{I}} u_i(q_i, q_i^U, q_i^N), \end{aligned} \tag{3}$$

where I and \bar{I} are sets of indices of active and inactive criteria, respectively, and q_i^U and q_i^N are the Utopia and approximation of the Nadir values, respectively. One can easily show that the treatment of a criterion as an *inactive* one has a similar effect to selecting the corresponding aspiration level close to the approximation of the Nadir for that criterion. Note, that for all criteria being active the ASF defined by (3) is equivalent to that of (2).

Component achievement functions $u_i(\cdot)$ are strictly monotone – either decreasing for minimized or increasing for maximized criteria, respectively – functions of the objective vector component q_i with values

$$\begin{aligned} u_i(q_i^U, \cdot) &= 1 + \bar{\gamma}, & u_i(\bar{q}_i, \cdot) &= 1, & u_i(\underline{q}_i, \cdot) &= 0, \\ u_i(q_i^N, \cdot) &= -\underline{\gamma}, \end{aligned} \tag{4}$$

where $\bar{\gamma}$ and $\underline{\gamma}$ are given positive parameters. A typical approach of setting these parameters to be equal to 0.1 and 10, respectively does not always result in concavity of ASF. Therefore, in order to keep the ASF concave, these parameters are dynamically adjusted by MCMA in the way described by Granat and Makowski (1998).

The piece-wise linear component achievement functions u_i proposed in Wierzbicki (1986) are defined by (5) and by (6) for minimized and maximized criteria, respectively:

$$u_i(q, \bar{q}, \underline{q}) = \begin{cases} \zeta_i w_i (\bar{q}_i - q_i) + 1, & \text{if } q_i < \bar{q}_i, \\ w_i (\bar{q}_i - q_i) + 1, & \text{if } \bar{q}_i \leq q_i \leq \underline{q}_i, \\ \eta_i w_i (q_i - \underline{q}_i), & \text{if } \underline{q}_i < q_i, \end{cases} \quad (5)$$

$$u_i(q, \bar{q}, \underline{q}) = \begin{cases} \zeta_i w_i (q_i - \underline{q}_i), & \text{if } q_i < \underline{q}_i, \\ w_i (\bar{q}_i - q_i) + 1, & \text{if } \underline{q}_i \leq q_i \leq \bar{q}_i, \\ \eta_i w_i (\bar{q}_i - q_i) + 1, & \text{if } \bar{q}_i < q_i, \end{cases} \quad (6)$$

where $w_i = 1/(q_i - \bar{q}_i)$, and ζ_i, η_i ($i = 1, 2, \dots, n$) are given parameters that are set in such a way that u_i takes the values defined by (4).

However, in order to allow for specification of not only required aspiration and reservations levels, but also for optional, finer, specification of preferences for the criteria values between aspiration and reservation levels, the ISAAP supports specification of the component achievement functions in a more general form than that of Eqs. (5) and (6). For this purpose, the CAF is defined as a piece-wise linear function u_i composed of segments u_{ji} defined by

$$u_{ji} = \alpha_{ji} q_i + \beta_{ji}, \quad q_{ji} \leq q_i \leq q_{j+1,i}, \quad (7)$$

$$j = 1, \dots, p_i,$$

where p_i is a number of segments for i -th criterion. Such a function for a minimized criterion is illustrated in Fig. 1. The thin line corresponds to a function composed of three segments, which are defined by four points, namely U, A^1, R and N , that correspond to the Utopia, aspiration, reservation and Nadir points, respectively. The solid line represents a modified function for which the previously defined aspiration level A^1 was moved to the point A , and two more points P^1 and P^2 were interactively defined.

Practical applications show that sometimes it is useful to set $\bar{q}_i = q_i^U$ and/or $\underline{q}_i = q_i^N$. Therefore, in order to also handle component achievement functions composed of only one segment (in cases

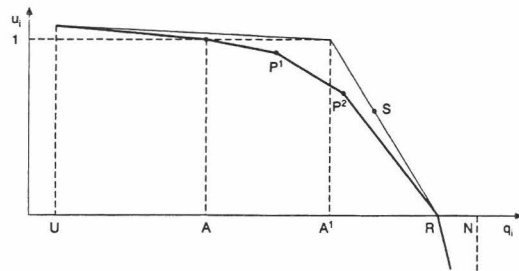


Fig. 1. Illustration of the piece-wise linear component achievement function for a minimized criterion.

when an aspiration level is set to the Utopia value and a reservation level is equal to an approximation of Nadir), ISAAP allows for $p_i \geq 1$.

The coefficients defining the segments are given by

$$\alpha_{ji} = \frac{u_{j+1,i} - u_{ji}}{q_{j+1,i} - q_{ji}}, \quad (8)$$

$$\beta_{ji} = u_{ji} - \alpha_{ji} q_{ji}, \quad (9)$$

where points (u_{ji}, q_{ji}) are interactively defined with the help of ISAAP. Concavity of the piece-wise linear functions $u_i(q_i)$ defined by segments (7) can be assured by a condition:

$$\alpha_{1i} > \alpha_{2i} > \dots > \alpha_{p_i i}, \quad j = 1, \dots, p_i. \quad (10)$$

Note that the component achievement functions u_i defined by (7) take the same form for minimized and maximized criteria. However, one should add – in addition to the condition (10) that assures concavity – a condition:

$$\alpha_{1i} < 0, \quad i \in I^{\min}, \quad (11)$$

$$\alpha_{p_i i} > 0, \quad i \in I^{\max}, \quad (12)$$

where I^{\min} and I^{\max} are sets of indices of criteria that are minimized and maximized, respectively. Conditions (11) and (12) are fulfilled automatically for the component achievement functions u_i , specified with the help of ISAAP.

A goal-type criterion can be used when a distance from a given target value (which can be changed during the interaction) is to be minimized. For such a criterion, a component achievement

function is composed of two parts: the first part is defined for the criterion values smaller than the target value; and the second part is defined for the criterion values larger than the given target. Such a function is illustrated in Fig. 2. The conditions specified above for maximized and minimized criteria hold for the first and second function, respectively. There is obviously only one point i , for which $\alpha_{i-1,i} > 0$ and $\alpha_{i,i+1} < 0$ and the criterion value for such a point corresponds to a target value (denoted by T) for a goal-type criterion. The function shown in Fig. 2 is symmetric but for many applications an asymmetric function is appropriate and therefore both types of functions for goal-type criteria are supported by ISAAP.

Both ISAAP and MCMA distinguish between the criterion type and status. The type of a criterion is defined during the preparatory stage (see Section 5.2) and cannot be changed during the interaction. However, quite often the user wants to temporarily treat a criterion in a different way. This can be achieved by changing the status of the criterion. The default status of the criterion means that a criterion is treated according to its type as originally defined by the user. The user may freely change the status of a criterion to one of the following: stabilized, inactive, disregarded and/or restore its default, i.e. originally defined, type.

Various GUIs can be used for specification of aspiration and reservation levels, as well as interpretation of solutions. In the approach applied in MCMA, the specified aspiration and reservation levels with optionally specified points between aspiration and reservation levels are used together

with the Utopia and an approximation of the Nadir point, both computed by MCMA to generate the component achievement functions (5) and (6). This is an easy and natural way to specify the desired values of each criterion by a corresponding aspiration level, and to scale trade-offs between criteria by corresponding pairs of aspiration and reservation values. Typically, initial aspiration values are far from being attainable, and the user has to modify her/his preferences, which are expressed by pairs of aspiration and reservation levels, in order to achieve solutions that are not too far away from the realistic goals.

The GUI uses the projection of values of the component achievement function on the interval $[0, 1]$. Such graphical presentation not only allows for specification of the user preferences, but also helps him/her in interpreting the solutions. This analysis can be done by projections of multi-dimensional criteria space into two-dimensional spaces composed for each criterion of its values and the degree of satisfaction of meeting preferences expressed by aspiration and reservation levels.

4. Structure and functions of MCMA and ISAAP

A DSS has to be problem specific. However, the reuse of developed software is a rational way for implementation of new applications. Therefore, typically a model-based DSS is composed of a number of mutually linked modules (see Makowski (1994b), for a more detailed discussion).

This Section provides an outline of the structure of MCMA, a software package, which has been developed for analysis of Linear Programming (LP) and Mixed Integer Programming (MIP) models using the ARBDS. A typical configuration of a DSS is illustrated in Fig. 3 for the application of the MCMA to the Land Use for Sustainable Agricultural Development Planning documented by Fischer et al. (1998). One should note that the structure of the DSS developed for Regional Water Quality Management for the Nitra River Basin by Makowski et al. (1996) differs only in the type of optimization problem (MIP instead of LP) and in the solver used.

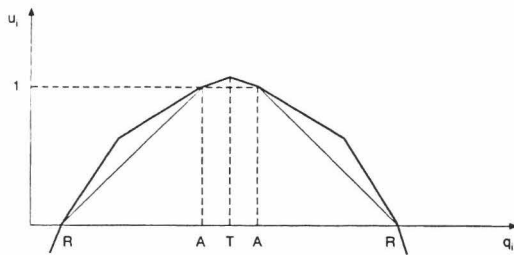


Fig. 2. Illustration of the piece-wise linear component achievement function for goal-type and for stabilized criteria.

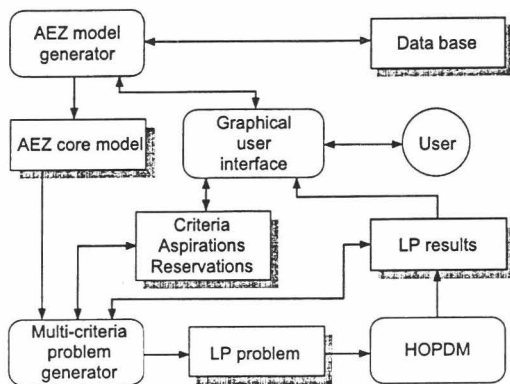


Fig. 3. The structure of a Decision Support System for the Land Use for Sustainable Agricultural Development Planning.

The MCMA is composed of a number of modular and portable software tools that are characterized below with brief descriptions of their functions:

- A GUI, which handles all the interaction with the user. GUI hides the differences between the modules of the DSS from the user by providing a uniform way of interaction with all the components of the DSS.
- A problem-specific core model generator (such as documented in Fischer et al. (1998), and in Makowski et al. (1996), for land use and water quality management problems, respectively) for generating the core model, which relates the decision variables with the resulting outcome variables. It is important to stress that the core model includes only physical and logical relationships, and not the preferential structure of the user. A more detailed discussion on core model specification is provided in Makowski (1994b).
- The ISAAP described in this paper which supports specification of user preferences both in terms of aspiration/reservation levels and, optionally, more precise specification in terms of the piece-wise linear component achievement function. ISAAP also provides the user with other means of control over problem analysis by allowing the changing of the criteria status, the selection of displayed solutions, etc. In terms

used in Fig. 3, the ISAAP is used for the definition of Aspirations, Reservations and for changing the status of Criteria. However, the ISAAP provides more functions than can be outlined in Fig. 3.

- The LP-Multi, described by Makowski (1994b), a C++ template class library for handling multi-criteria problems using the methodology outlined in Section 2. The resulting optimization problem (either LP or MIP) is based on the core model and on a current representation of the preferential structure of the user, which is expressed by the aspiration and reservation levels, and optionally also by additional segments of CAF.
 - Two solvers for solving LP and MIP optimization problems, respectively. The solvers are robust because in a typical application their use is hidden from the user. Therefore, solvers used in a DSS must not require interaction with the user. In MCMA, an appropriate solver is selected automatically depending on the type of the core model (LP or MIP). The HOPDM, interior-point based LP solver, in the version documented by Gondzio and Makowski (1995), is used for LP models, while the MOMIP, described by Ogryczak and Zorychta (1996), is applied to mixed integer models.
 - A data interchange tool LP-DIT described in Makowski (1994a). This tool provides an easy and efficient way for the definition and modification of LP and MIP problems, as well as the interchange of data between a problem generator, a solver, and other software modules that support problem modification and solution analysis. LP-DIT is used for the definitions of the core model and of the auxiliary LP or MIP optimization problems defined for each multi-criteria problem, as well as for handling the optimization results.
- The portability of the developed tools is achieved by using C++ programming language and a commercial tool for development of the portable GUI, namely zApp library (Inm, 1995). Modular structure and portability allow for the reuse of most of the components needed for a DSS applied to other problems. It also facilitates experiments with different solvers and with modules providing

problem specific interaction with the user. Note, that a new application only requires the development of a model generator and, optionally, a problem-specific module for a more detailed analysis of the results.

ISAAP is a module that can be used as a part of a model-based DSS using a multi-criteria model analysis using aspiration and reservation levels. ISAAP plays a central role in the interaction with the user by providing all the functions necessary for interactive analysis of solutions and for specification of a new multi-criteria optimization problem, namely:

- Specification of an aspiration and a reservation level.
- Optional specification of a piece-wise linear membership function for criteria values between aspiration and reservation levels.
- Changing the status of a criterion by stabilizing it (minimizing a deviation of the criterion value from a given target value) or temporarily disregarding it.
- Supporting analysis of previously computed solutions by keeping record of the solution's history.

ISAAP has been designed and implemented with an inexperienced user in mind. Therefore the use of ISAAP by a person familiar with a window system is easy and does not require substantial amounts of training. Moreover, on-line documentation and tutorial example, outlined in Sections 5.6 and 6, respectively, substantially help using MCMA.

ISAAP has also been applied as a part of other DSSs to case studies that require analysis of non-linear models (see Section 7) without changing a single line of code. This illustrates clearly the power of modular tools.

5. Multi-criteria model analysis with ISAAP

Currently, the ISAAP module is available only as a part of the MCMA. Therefore, this section provides the following three areas of information (assembled in the following subsections) that are of interest to any user of ISAAP and that correspond to stages of model preparation and analysis:

- Preparation of a core model.
- The preparatory stage of model analysis.
- Information about the interaction with ISAAP, which is useful for the user who is not interested in the technical details of the implementation of ISAAP.

This section provides an overview of selected functions of ISAAP. The complete description of ISAAP is provided by Granat and Makowski (1998).

5.1. Preparation of a core model

As already discussed in Section 2, the first stage of a problem analysis is building a corresponding *core model*. Two core models are distributed with this software, one of LP and the other of MIP type. These models are taken from the real-world applications, namely:

- Optimizing Land Use for Sustainable Agricultural Development Planning documented in Fischer et al. (1998).
- Regional Water Quality Management Problem, case study of the Nitra River Basin (Slovakia) documented in Makowski et al. (1996).

The first of these models is used in the illustrative examples presented in Section 6. However, it should be stressed that MCMA does not support the specification, verification or modification of a core model. In order to be properly used in MCMA, the core model has to conform to a number of requirements discussed in more detail in Makowski (1994b). The preparation of a core model for a non-trivial problem requires not only knowledge and experience, but also software tools. Such tools can be either a problem-specific model generator, see Makowski (2000), for an example; or a general purpose modeling tool, like AMPL (Fourer et al., 1996) or GAMS (Brooke et al., 1992).

The core model to be analyzed by MCMA and ISAAP must be provided in the form of LP-DIT binary file *id.cor*, where *id* is the implied problem name. The tutorial models distributed with MCMA are already in the LP-DIT format. For LP and MIP type models, many of the modeling tools can generate a model in the MPS format.

Models available in the MPS format can be converted by the LPDIT utility distributed with the MCMA.

5.2. Preparatory stage of a model analysis

After a preparation of a core model one has to define a set of criteria that will be used for the model analysis. For this purpose one can apply a file which contains a set of predefined criteria. It is also possible to interactively select (out of all core model variables) the outcome variables that will serve as criteria. Subsequently, one selects for each criterion its name, type and units in which criterion values are expressed. After the definition of the criteria, MCMA starts automatically the computation of the pay-off table composed of the Utopia and approximations of Nadir point. The last step in the preparatory stage is computation of the compromise solution, obtained by automatically setting the aspiration and reservation levels to the Utopia and Nadir points, respectively.

It is also possible to continue an analysis previously done by ISAAP. For this purpose, MCMA stores information about the analysis done so far and the user can select an instance of the model to be analyzed from the main menu of MCMA, instead of performing the actions described in this and in the previous subsection.

5.3. Basic interaction with ISAAP

After the preparatory stage outlined in Section 5.2 is completed one can start the interaction with ISAAP. The main ISAAP window (shown in Fig. 4, see Section 6) provides access to a number of functions that can be selected from the main menu of the ISAAP. The most commonly used, and therefore the default, function is described in this section while more advanced functions are discussed in Section 5.5.

5.3.1. Specification of aspiration and reservation levels

The default function of ISAAP is a selection of aspiration and reservation levels that define the

corresponding component achievement function for each active criterion. For each criterion, the projection of the last specified component achievement function $u_i(q_i)$ on the interval $[0, 1]$ is plotted.

The aspiration and reservation levels can be set by using either the mouse or a keyboard. Clicking the mouse results in moving the nearest aspiration or reservation point to the area where the mouse is currently pointing. This is the easiest and quickest way of setting these values. The exact values for aspiration and reservation levels can be displayed and updated from a pop-up window, which can be created by selecting the Values item from the ISAAP menu.

Aspiration and reservation levels must not be too close (see Makowski, 1994b, for more detailed information). Should the specified values for those levels for a criterion be close, then either they should be modified or the criterion status has to be changed for the stabilized one. If such a case is detected, the user is asked to either change the aspiration and/or reservation levels, or the status of the criterion.

5.4. Pareto-optimal solution

After the user has finished the specification of her/his preferences, a parametric optimization problem is generated by MCMA. The solution of this problem provides a Pareto-optimal solution, which is closest to the specified aspiration levels.

In the current implementation of MCMA, one of the following two solvers is used for solving such optimization problem:

- HOPDM (see Gondzio and Makowski, 1995), interior point based LP solver, especially efficient for medium and large scale problems.
- MOMIP (see Ogryczak and Zorychta, 1996), modular optimizer for MIP.

The default selection of the solver is done by MCMA. HOPDM is chosen for LP optimization problems and MOMIP for MIP problems. The set of solver parameters is selected in order to allow for efficient solving of a broad class of respective types of optimization problems. The user is advised to contact the authors should the solution time become unacceptable. A problem specific

tuning of solver parameters may substantially improve the performance of a solver.

5.5. Advanced interaction with ISAAP

The user can specify his/her preferences for the criteria values between aspiration and reservation levels by specification of a piece-wise linear function. In order to create or delete additional points, a corresponding item (Add a point or Delete a point, respectively) from the Shape main menu should be selected. The points are added or deleted (depending on the selection made from the Shape main menu) by clicking a mouse at the place, where a point should be added/deleted. Adding/deleting points is continued until another selection is made from the Shape menu. In order to come back to the moving points mode of the interface, one should select the Move a point from the same menu. The background of the ISAAP window is changed to green or red for adding and deleting points modes, respectively. Additional points can be later moved by clicking a mouse in the way similar to modification of aspiration and reservation levels.

Note that the concavity conditions by Eqs. (10)–(12) can be forced by ISAAP only for criteria that do not have additional points. By moving additional points the user may specify a non-concave function which does not fulfill these conditions. In such a case, ISAAP informs the user about the points causing non-concavity of the function and ask the user to modify the corresponding function in order to make it concave.

5.5.1. Goal-type and stabilized criteria

Goal-type criterion and criterion whose status has been changed to stabilized are treated in the same way, with only one exception: for a goal-type criterion the user can specify any target value, whereas for a stabilized criterion, a target value must lay somewhere between the corresponding Utopia and Nadir components. Hence, we will also use the term stabilized for goal-type criteria in this subsection. For the sake of brevity, we will ignore the index i of a criterion q_i and of the corresponding quantities – target

value, aspiration, reservation levels – in this subsection.

For a stabilized criterion one has to select a target (desired) value \bar{q} and to specify two pairs of aspiration and reservation levels, which are interpreted as *still accepted* and *no longer accepted* values of the criterion. One pair is defined for a surplus (over the target value \bar{q}), while the second is for a deficit (values lower than \bar{q}). The pairs of aspiration and reservation levels are denoted by $(\bar{q}^+, \underline{q}^+)$ and $(\bar{q}^-, \underline{q}^-)$, respectively.

For many criteria the deviations from the target value in both surplus and deficit directions have similar meaning, therefore it is useful to distinguish cases in which a target value is equal to the mean of aspiration levels. This type of a stabilized criterion is called the *symmetric* type and the following condition is forced by the way in which the interaction is implemented:

$$\bar{q} = (\bar{q}^+ + \underline{q}^+)/2 = (\bar{q}^- + \underline{q}^-)/2. \quad (13)$$

The *symmetric* type is the default type for a stabilized criterion. It can be changed from the Shape menu to the *general* type for a criterion in which the user does not accept the condition (13) and wants to specify aspiration and reservation levels independently for surplus and deficit. The *general* type of a stabilized criterion provides much more flexibility at the expense of a more time consuming interaction. Note that one can optionally define a piece-wise linear CAF for both (*symmetric* and *general*) types of a stabilized criterion.

The target value for a goal or stabilized criterion can be changed with the mouse (by changing the interaction mode through selection of the Move a target from the Shape main menu; in this mode, the background of the ISAAP window is changed to a yellow color) or from the keyboard (using the dialog activated by selection of the Values item from the main ISAAP menu).

Processing stabilized criteria is implemented by the generation of additional criteria. This is done by MCMA in a way that is transparent to the user. Details of the processing are documented in Makowski (1994b).

5.5.2. Status of criteria

The user can easily change the status of a criterion to one of the following:

min/max/goal – a criterion has its default status (either minimized, or maximized or of the goal type).

stabilized – a minimized or maximized criterion is converted as a stabilized criterion.

inactive – a criterion is temporarily disregarded and its component achievement function is not defined, however, the criterion still enters the scalarizing function (3).

disregarded – a criterion is completely dropped from entering the scalarizing function. This feature has been found very useful for a more detailed analysis of more complex problems, especially when more than four to five criteria are considered.

5.5.3. History of solution

ISAAP keeps record of all the Pareto-optimal points and the corresponding aspiration and reservation levels. The history of all solutions can be examined not only on plots but also in the form of a spreadsheet. The selected M solutions are displayed in the main ISAAP window. The previous solutions that belong to the set of displayed solutions are marked by small squares with labels. If the next to last solution does not belong to this set, then it is marked by a small green circle without a label. The last solution is always marked by a white circle without a label and it is connected with the previous solution by a thin line.

Solutions can be added to or removed from the set of displayed solutions by using a dialog. The last (rightmost) field in the solution spreadsheet provides space for user comments or notes related to a corresponding solution.

5.6. On-line help

The documentation of MCMA and ISAAP has been written in LaTeX using additional commands that has been defined in order to make it possible to automatically prepare two electronic versions of this document. The electronic versions can be viewed by one of the two browsers that can be

invoked from MCMA, namely *Netscape* and *zHelp*. Such an approach has several advantages:

- The on-line help is based on an automatically generated electronic version of the corresponding documentation, therefore it is easy to keep the on-line help consistent with a hard copy version of the documentation.
- The on-line help can be viewed by *Netscape*, which is commonly used on both MS-Windows and Unix installations, and/or by *zHelp*, the portable browser distributed with MCMA. Due to the limitations of *zHelp*, which does not support the full implementation of the HTML, the functionality of the corresponding version of the on-line help is slightly limited.
- Additional commands for LaTeX define labels that are automatically associated with corresponding pages of the on-line help. These associations are converted into a dictionary, which is distributed with the on-line help. This makes it possible to implement a context-sensitive help, i.e., controlling loading of appropriate pages by the software. However, the user can also navigate independently through the electronic version of the documentation.

6. Example of analysis by ISAAP module

Due to space limitations, this section provides only an illustration of basic interaction with ISAAP, which starts after a compromise solution is computed, or after a continuation of the analysis has been requested. The complete tutorial for using ISAAP can be found in Granat and Makowski (1998) or in the on-line documentation distributed with MCMA.

For the Land Use Planning example presented in this section, four criteria have been selected (see Section 5.2), namely average food production, land use, net revenue and total erosion, denoted by *FoodAv*, *Land*, *NetRev* and *TotEro*, respectively. In Fig. 4 the values of criteria for the compromise solution are marked by white circles. A compromise solution is typically not acceptable, because the aspiration and reservation levels are set automatically by MCMA to be equal to components of the Utopia and Nadir points, respec-

tively. However, such a solution is a good starting point for analysis of the model. Therefore, let us assume that the user wants to primarily improve the level of food production and therefore she/he attempts to achieve this by increasing the reservation level for the corresponding criterion to about 300, without changing aspiration and reservation levels for other criteria (see Fig. 4 for the illustration). This can be achieved by a sequence of clicking the mouse to points corresponding to a new reservation level. A reader will probably be surprised why this cannot be achieved by a single click. The reason for this comes from the results of experiments on how the interaction should be organized. It seems that the easiest way to move points is by clicking the mouse to a new desired value for the point that is nearest to the new desired value. In this case, we would like to move the reservation level of the criterion FoodAv, which was below 100, therefore we should first click a value around 200, then 250, and finally 300. This description looks complicated, but these steps quickly become quite natural.

The Pareto-optimal solution that corresponds to the preferences set in Fig. 4 is illustrated in Fig. 5. We have indeed succeeded in considerably improving the food production level. However, this results in the substantial increase of land use

and of erosion. Note, that the level of net revenue has not changed substantially, but our satisfaction level (measured by the value of the CAF function) has decreased. This decrease corresponds to the fact that the current solution is more distant from the aspiration level than the compromise solution.

Before making further steps in analyzing the tutorial model, let us explain a few basic rules applied to the organization of the interaction. Unfortunately, illustration of some of these rules is difficult to see in black and white printed figures; however, these figures should help the reader to reproduce the results on her/his color screen.

For each of the defined criterion, the last computed value of each criterion is marked by a small white circle. One iteration consists of analysis of previous solutions, selection of new aspiration and reservation levels, optional definition of additional segments of piece-wise linear functions defining the component achievement functions, optional change of criteria status, and computation of a new Pareto-optimal solution.

Once you have decided the status of all criteria and the new set of aspiration and reservation levels for each criterion, select the Pareto sol. item from the MCMA main menu. This will start the generation and solution of the new optimization problem. The ISAAP window is minimized during

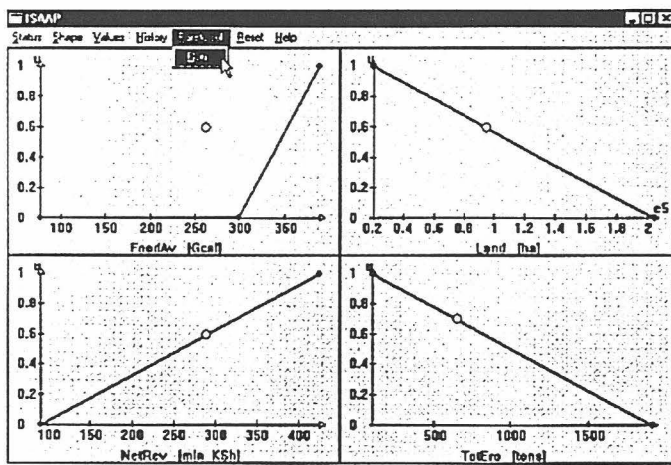


Fig. 4. Increasing the reservation level for FoodAv.

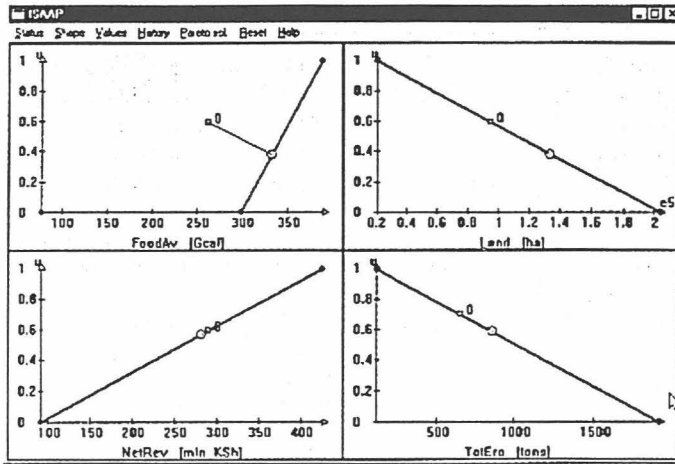


Fig. 5. Solution for preferences set in Fig. 4.

optimization and it is restored to its previous size after a new solution is ready for analysis. We will illustrate these options by the following iterations discussed below.

After the first attempt to improve only one criterion, let us assume that we would like to now keep the aspiration and reservation level for the FoodAv criterion without changes, and we would

also like to decrease the erosion. In order to make our current goals more realistic, we agree to use more land and would be satisfied with the current level of revenue, but would not like it to drop below 150. Fig. 6 illustrates the setting of the aspiration and reservation levels that may represent such preferences. The corresponding Pareto-optimal solution is presented in Fig. 7. We can

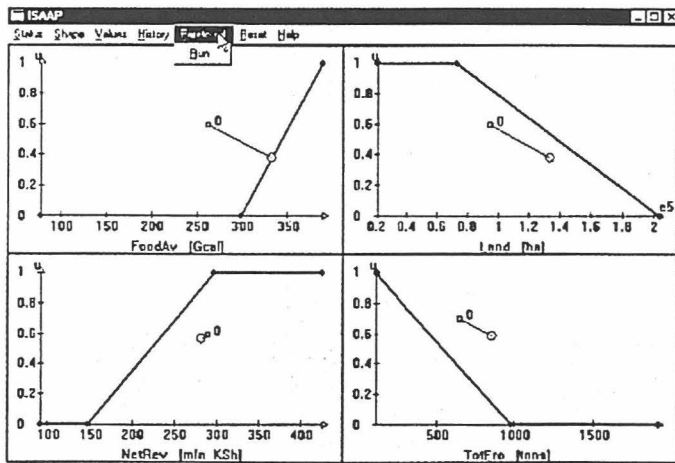


Fig. 6. Setting of preferences in the second iteration.

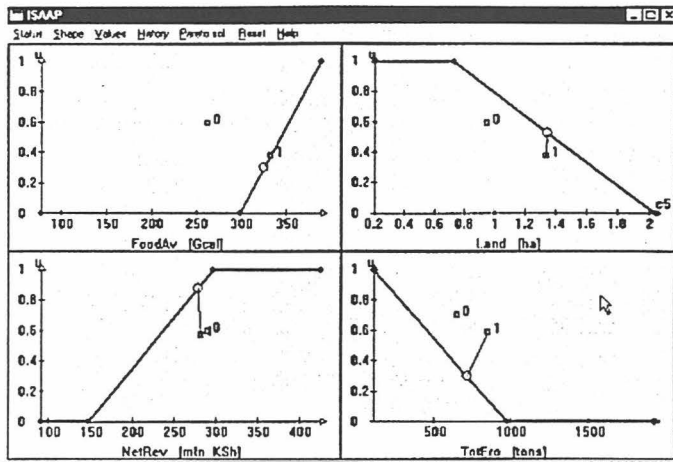


Fig. 7. Results of the second iteration.

observe that we have achieved only a small improvement for the TotEro criterion without much change in the values of the other three criteria. The decrease of the satisfaction level for the TotEro criterion might be surprising, because the value of this criterion has improved along with the decreased erosion. However, explanation for this phenomenon is easy, if one

remembers the interpretation of the CAF, which represents the satisfaction level. In the compromise solution the satisfaction for this criterion was much higher because the reservation level was also much higher, therefore the value of this criterion in the compromise solution remarkably exceeds our expectations. However, in the second iteration, we increased our expectations consid-

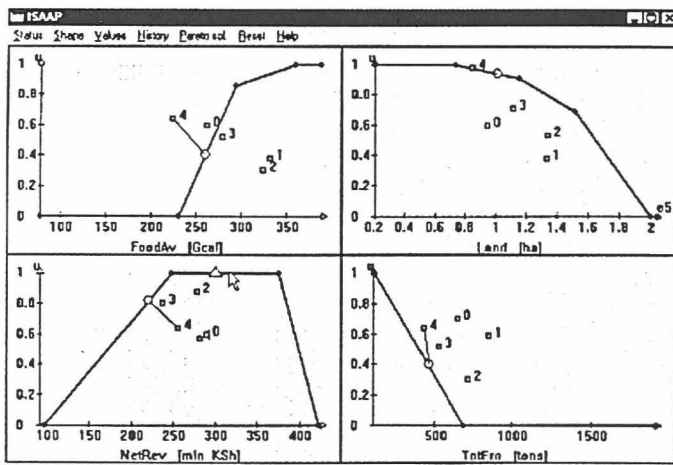


Fig. 8. Solution for the fifth iteration.

erably by moving the reservation level to the value of 1000; therefore, the value of this criterion for the new solution is much less satisfactory (taking into account the new preferences) than in the compromise solution.

By now it becomes clear that we cannot decrease the erosion much further without relaxing our expectations for the other criteria, especially for revenue and food production.

We will conclude this short presentation of ISAAP by showing in Fig. 8 a screen illustrating a more advanced stage of analysis after the fifth iteration. Six solutions are presented in this figure: the compromise solution labeled by 0, four previous solutions labeled by numbers 1–4, and the last solution marked by white circles. One can see that the CAF for two of the criteria take forms of piece-wise linear functions for the criteria values between the corresponding aspiration and reservation levels. The criterion *NetRev* has been stabilized using non-symmetric shape of the CAF, obviously appropriate in this case. A reader interested in a complete tutorial for using ISAAP is advised to consult Granat and Makowski (1998).

7. Conclusions

The paper describes a methodology of interactive specification of user preferences applied in the MCMA software package supporting multi-criteria model analysis.

The authors have received many comments and suggestions from users of MCMA. These have helped considerably in improving the MCMA prototypes distributed earlier. The authors have also been involved in the implementations of several applications of MCMA, including those listed below. From these experiences the authors can conclude that the approach presented in this paper and its implementation have the following main advantages for the user:

- provide simple, intuitive and efficient user interface,
- significantly speed up the process of model analysis,
- support a truly interactive analysis of also large LP and MIP models.

Until now, the MCMA has been implemented within the following applications:

- A DSS developed for the Regional Water Quality Management Problem, a case study of the Nitra River Basin (Slovakia) documented by Makowski et al. (1996).
- Multi-criteria Analysis in Optimizing Land Use for Sustainable Agricultural Development Planning described by Antoine et al. (1997). This application is a result of the cooperation between IIASA and the FAO (Food and Agriculture Organization of the United Nations).
- Multi-criteria Analysis of Urban Land-Use Planning, described by Matsuhashi (1997).
- A number of engineering applications in mechanics, automatic control and ship navigation summarized in Wierzbicki et al. (2000).

Additionally, over 300 copies of MCMA package have been made available for research and educational purposes.

The MCMA software is available from: <http://www.iiasa.ac.at/~marek/soft> free of charge for non-commercial research and educational purposes (please read carefully the license agreement available on the same Web page). The distributable set also contains two solvers (HOPDM and MOMIP for LP and MIP problems, respectively) and two core models (corresponding to the Nitra and Land Use case studies mentioned above). Therefore, MCMA can be easily used for analysis of any LP or MIP model provided that a corresponding *core model* is available in the LP-DIT format (Makowski, 1994a) or in the MPS format.

Acknowledgements

The authors of the paper acknowledge and appreciate the contribution of Mr Grzegorz Wójcik of the Warsaw University of Technology, who has developed a tool for converting a single source documentation in the form of LaTeX (with additional styles) into three types of documents, namely a standard LaTeX document, the HTML files, and the restricted HTML files accepted by a portable *zHelp* utility. The tool also generates a

dictionary which makes it possible to implement a context-sensitive help in a C++ program.

References

- Antoine, J., Fischer, G., Makowski, M., 1997. Multiple criteria land use analysis. *Applied Mathematics and Computation* 83 (2/3), 195–215. Available also as IIASA's RR-98-05.
- Brooke, A., Kendrick, D., Meeraus, A., 1992. GAMS, A User's Guide, release 2.25. The Scientific Press, Redwood City.
- Charnes, A., Cooper, W., 1967. *Management Models and Industrial Applications of Linear Programming*. Wiley, New York.
- Fischer, G., Granat, J., Makowski, M., 1998. AEZWIN an interactive multiple-criteria analysis tool for land resources appraisal. Interim Report IR-98-051. International Institute for Applied Systems Analysis, Laxenburg, Austria. Available on-line from <http://www.iiasa.ac.at/~marek/pubs>.
- Fourer, R., Gay, D., Kernighan, B., 1996. *Using AMPL Plus*. Compass Modeling Solutions Inc., San Francisco.
- Gardiner, L., Steuer, R., 1994. Unified interactive multiple objective programming. *European Journal of Operational Research* 74, 391–406.
- Gondzio, J., Makowski, M., 1995. HOPDM, modular solver for LP problems, user's guide to version 2.12. Working Paper WP-95-50. International Institute for Applied Systems Analysis, Laxenburg, Austria. Available on-line from <http://www.iiasa.ac.at/~marek/pubs>.
- Granat, J., Makowski, M., 1998. ISAAP – Interactive Specification and Analysis of Aspiration-based Preferences. Interim Report IR-98-052. International Institute for Applied Systems Analysis, Laxenburg, Austria. Available on-line from <http://www.iiasa.ac.at/~marek/pubs>.
- Haimes, Y., Hall, W., 1974. Multiobjectives in water resource systems analysis: The surrogate trade-off method. *Water Resources Research* 10, 615–624.
- Inm, 1995. *zAPP, The Portable C++ Application Framework, Programmer's Guide*.
- Isermann, H., Steuer, R.E., 1987. Computational experience concerning payoff tables and minimum criterion values over the efficient set. *European Journal of Operational Research* 33, 91–97.
- Korhonen, P., Wallenius, J., 1989. Observations regarding choice behaviour in interactive multiple criteria decision-making environments: An experimental investigation. In: Lewandowski, A., Stanchev, I. (Eds.), *Methodology and Software for Interactive Decision Support*. Lecture Notes in Economics and Mathematical Systems, vol. 337. Springer, Berlin, pp. 163–170.
- Lewandowski, A., Wierzbicki, A. (Eds.), 1989. *Aspiration Based Decision Support Systems: Theory, Software and Applications*. Lecture Notes in Economics and Mathematical Systems, vol. 331. Springer, Berlin.
- Makowski, M., 1994a. LP-DIT, Data Interchange Tool for Linear Programming Problems, version 1.20. Working Paper WP-94-36. International Institute for Applied Systems Analysis, Laxenburg, Austria. Available on-line from <http://www.iiasa.ac.at/~marek/pubs>.
- Makowski, M., 1994b. Methodology and a modular tool for multiple criteria analysis of LP models. Working Paper WP-94-102. International Institute for Applied Systems Analysis, Laxenburg, Austria. Available on-line from <http://www.iiasa.ac.at/~marek/pubs/>.
- Makowski, M., 2000. Modeling paradigms applied to the analysis of European air quality. *European Journal of Operational Research* 122, this issue.
- Makowski, M., Somlyódy, L., Watkins, D., 1996. Multiple criteria analysis for water quality management in the Nitra basin. *Water Resources Bulletin* 32 (5), 937–951.
- Matsuhashi, K., 1997. Application of multi-criteria analysis to urban land-use planning. Interim Report IR-97-091. International Institute for Applied Systems Analysis, Laxenburg, Austria.
- Nakayama, H., 1994. Aspiration level approach to interactive multi-objective programming and its applications. Working Paper WP-94-112. International Institute for Applied Systems Analysis, Laxenburg, Austria.
- Ogryczak, W., Lahoda, S., 1992. Aspiration/reservation-based decision support – a step beyond goal programming. *Journal of Multi-Criteria Decision Analysis* 1 (2), 101–117.
- Ogryczak, W., Zorychta, K., 1996. Modular optimizer for mixed integer programming, MOMIP version 2.3. Working Paper WP-96-106. International Institute for Applied Systems Analysis, Laxenburg, Austria.
- Sawaragi, Y., Nakayama, H., Tanino, T., 1985. *Theory of Multiobjective Optimization*. Academic Press, New York.
- Steuer, R., 1986. *Multiple Criteria Optimization: Theory, Computation, and Application*. Wiley, New York.
- Wierzbicki, A., 1977. Basic properties of scalarizing functionals for multiobjective optimization. *Mathematische Operationsforschung und Statistik Optimization* 8, 55–60.
- Wierzbicki, A., 1980. The use of reference objectives in multiobjective optimization. In: Fandel, G., Gal, T. (Eds.), *Multiple Criteria Decision Making, Theory and Applications*. Lecture Notes in Economics and Mathematical Systems, vol. 177. Springer, Berlin, pp. 468–486.
- Wierzbicki, A., 1986. On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR Spektrum* 8, 73–87.
- Wierzbicki, A., Makowski, M., Wessels, J. (Eds.), 2000. *Model-Based Decision Support Methodology with Environmental Applications*. Kluwer Academic Publishers, Dordrecht.

