Lecture Notes
in Economics and
Mathematical Systems

Managing Editors: M. Beckmann and W. Krelle

351

A. Lewandowski V. Volkovich (Eds.)

Multiobjective Problems of Mathematical Programming

Proceedings, Yalta, USSR, 1988

Springer-Verlag

Lecture Notes
in Economics and
Mathematical Systems

Managing Editors: M. Beckmann and W. Krelle

351

A. Lewandowski V. Volkovich (Eds.)

Multiobjective Problems of Mathematical Programming

Proceedings of the International Conference on
Multiobjective Problems of Mathematical Programming
Held in Yalta, USSR, October 26–November 2, 1988

Springer-Verlag
Berlin Heidelberg New York London
Paris Tokyo Hong Kong Barcelona

Editorial Board

H. Albach M. Beckmann (Managing Editor)
P. Dhrymes G. Fandel G. Feichtinger W. Hildenbrand W. Krelle (Managing Editor)
H. P. Künzi K. Ritter U. Schittko P. Schönfeld R. Selten

Managing Editors
Prof. Dr. M. Beckmann
Brown University
Providence, RI 02912, USA

Prof. Dr. W. Krelle
Institut für Gesellschafts- und Wirtschaftswissenschaften
der Universität Bonn
Adenauerallee 24–42, D-5300 Bonn, FRG

Editors
Dr. Andrzej Lewandowski
Former Leader of the MDA Project
IIASA now affiliated with Wayne University
Dept. of Electrical and Computer Engineering
Detroit, Michigan 48202, USA

Prof. Dr. Victor Volkovich
Deputy Director of the V. Glushkov Institute of
Cybernetics of the Ukrainian Academy of Sciences
Kiev, USSR

ISBN 3-540-53432-6 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-53432-6 Springer-Verlag New York Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in other ways, and storage in data banks. Duplication of this publication or parts thereof is only permitted under the provisions of the German Copyright Law of September 9, 1965, in its current version, and a copyright fee must always be paid. Violations fall under the prosecution act of the German Copyright Law.

© International Institute for Applied Systems Analysis, Laxenburg/Austria 1991
Printed in Germany

Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
2142/3140-543210 – Printed on acid-free paper

Preface

These Proceedings report the scientific results of the International Conference on *Multiobjective Problems of Mathematical Programming* organized by the System and Decision Sciences Program at IIASA, the Ukrainian Academy of Sciences (V. Glushkov Institute of Cybernetics) and the Committee for Systems Analysis of the U.S.S.R. Academy of Sciences and are devoted to the theory of multiobjective optimization, procedures for solving multiple objective mathematical programming problems, applied problems of multiobjective optimization and interactive and intelligent decision support systems. The Conference took place in Yalta, U.S.S.R., on the Black Sea coast.

More than 150 scientists scientists from the following countries: Austria, Belgium, Bulgaria, the People's Republic of China, Czechoslovakia, Finland, FRG, GDR, Italy, Japan, Poland, the U.S.A. and the U.S.S.R. participated in this Conference. The Conference is one of a series of meetings organized by IIASA with collaboration of scientific institutions from the National Member Organization countries. The previous meetings and conferences took place in Austria (1983), Hungary (1984) Germany (1985) and Bulgaria (1987). All proceedings of these meetings have been published by Springer Verlag in the series Lecture Notes in Economics and Mathematical Systems.

The research on decision support systems has a long tradition at IIASA. The Institute, being the forum for common research of scientists from East and West, with different cultural backgrounds and different experiences with real life applications of their results, operates an international network of scientific institutions involved in research related to the methodology of decision analysis and decision support systems. This Conference was an especially unique contribution to this cooperation, since it provided the first opportunity for people from the Western countries to learn about developments in the Soviet School of MCDM (Multiple Criteria Decision Making) and DSS (Decision Support Systems). This resulted in strengthening the East-West cooperation on MCDM, DSS and related topics.

The approach to research in Multiple Objective Decision Support, Multiple Criteria Optimization and related topics represented by IIASA and its collaborating institutions assumes a high level of synergy between three main components: methodological and theoretical backgrounds, computer implementation of decision support systems and real life applications. This synergy is reflected in the subjects of papers presented during the Conference as well as in the structure of the Proceedings which is divided in three main sections.

In the first section, *Theory and Methodology of Multiple Criteria Optimization*, 21 papers discussing new theoretical developments in multiple criteria optimization are presented. *Larichev* presents general methodology and unified approach to Multiple Criteria problems. *Mikhalevich* and others present new algorithms for solving Multiple Criteria problems and discuss their theoretical properties as well as implementation aspects. Especially interesting is a paper by *Liebermann* discussing the current state of the Multi-Objective programming in the U.S.S.R from the point of view of Western scientists.

In the second section, *Applications of Multiple Criteria Optimization*, 9 papers presents real-life applications of Multiple Criteria Optimization. These applications include water management problems (*Sukhorukov*), industrial applications (*Kopytowski*, *Serafini and others*), managerial decision making (*Spronk*, *Parizek*) as well as engineering design problems (*Voloshin*).

In the third section, *Multiple Criteria Decision Support*, 5 papers discuss the application of Multiple Criteria Optimization for development Decision Support Systems are presented. These papers present methodological aspects of Decision Support Systems (*Britkov*, *Petrovsky*, *Vetschera*) as well as practical implementations and applications (*Dobrowolski*, *Britkov*).

One of the important outcomes of the Conference were conclusions regarding further directions of research in Multiple Criteria Optimization, in particular, in the context of cooperation of scientists from Eastern and Western countries.

The editors of these Proceedings would like to thank IIASA for financing the Workshop and for its continuous support and encouragement for research in the field of Decision Support Systems. This support and encouragement came especially from Prof. Alexander Kurzhanski, Chairman of the System and Decision Sciences Program at IIASA. It would not have been possible to organize the Conference without the strong support from the Ukrainian Academy of Sciences of the U.S.S.R, the V. Glushkov Institute of Cybernetics and its Director, Academician V.C. Mikhalevich and the Committee for Systems Analysis of the U.S.S.R. It would not have been possible to organize this conference without the strong involvement of the Local Organizing Committee which included scientists from several institutions in the U.S.S.R. Finally, the editors would like to thank the authors for their participation in the Conference and permission to publish their contributions in this volume.

A. Lewandowski
V. Volkovich

Table of Contents

Part 2

Applications of Multiple Criteria Optimization

Part 1

Theory and Methodology of Multiple Criteria Optimization

The Use of the Qualitative Information on the Importance of Particular Criteria for the Computation of Weighting Coefficients

Dmitry I. Batishchev, Vladimir F. Anuchin, Dmitry E. Shaposhnikov
Gorky State University, Gorky, USSR

Abstract

The objective of this paper is to describe the procedure of the convolution of the vector optimality criterion by means of addition or generalized logical convolution in solving the multiobjective optimization problem.

The procedure of computing the weighting coefficients of relative importance with regard to the current value of the controlled parameters on the basis of qualitative information on the binary relations of preference for the particular criteria is shown, with the "guaranteed result" principle realized.

The proofs not supplied here can be obtained from the publications of the authors (see references).

1 Introduction

Multiobjective optimization problems have been widely used as one of the decision making models (Zionts, 1978). Taking particular criteria of optimality for linear functions a number of methods of numerical solutions of this type of problems are considered in (Lewandowski and Wierzbicki, 1988; Mikhalevich and Volkovich, 1982; Cohon, 1978). More general methods of vector criteria convolution are presented in (Germeier, 1971). The general criterion as the common value for particular criteria often involves nonnegative weighting coefficients, whose numerical values show the relative importance (preference) of the particular criteria. In some cases (Batishchev, 1984), the process of decision making requires considering the dependence of weighting coefficients on particular criteria in every point of the controlled parameters variation region. Assuming that the decision maker finds it difficult to calculate exactly the numerical values of the weighting coefficients, but is able to set the qualitative information on the binary relations of preference for certain particular criteria pairs, the setting of the weighting coefficients can be done on the basis of the "guaranteed result" principle (Germeier, 1971).

The decision of the above problem is presented in this paper.

2 The general outline of reducing the multiobjective problem to that of nonlinear programming

The starting point is a multiobjective optimization problem with s nonlinear particular optimal criteria obtained from the feasible set $\mathbf{D}$:

$$\min_{\vec{x}\in\mathbf{D}} Q_1(\vec{x}); \quad \ldots \quad ; \min_{\vec{x}\in\mathbf{D}} Q_s(\vec{x}). \tag{1}$$

The particular criteria $Q_i(\vec{x})$, $i = \overline{1,s}$ are supposed to have the same scale of measurement or to have been reduced to an immeasurable type.

Transition from partially formulated model of decision making problem (1) to conventional correct 'single-objective extremal problem' model is nothing else but the process of convolution of the optimality vector criterion $\vec{Q}(\vec{x}) = (Q_1(\vec{x}), \ldots, Q_s(\vec{x}))$ into a scalar function $W(\vec{x})$ presenting the generalized optimality criterion:

$$\min_{\vec{x}\in\mathbf{D}} W(\vec{\lambda}, Q_1(\vec{x}), \ldots, Q_s(\vec{x})) \tag{2}$$

where $\vec{\lambda} = (\lambda_1, \ldots, \lambda_s)$ are weighting coefficients of the relative importance of the particular optimality criteria.

Let us further consider as a generalized optimization criterion $W(\vec{\lambda}, \vec{Q}(\vec{x}))$ the following two types of convolution:

a) addition:

$$W(\vec{\lambda}, \vec{Q}(\vec{x})) = \sum_{i=1}^{s} \lambda_i Q_i(\vec{x}) \tag{3}$$

b) generalized logical convolution:

$$W(\vec{\lambda}, \vec{Q}(\vec{x})) = \max_{1\le i\le s}(\lambda_i Q_i(\vec{x})), \tag{4}$$

where

$$\lambda_i \geq 0; \qquad i = \overline{1,s}; \qquad \sum_{i=1}^{s} \lambda_i = 1. \tag{5}$$

If the decision maker finds it difficult to calculate exact numerical values of the weighting coefficients $\vec{\lambda}$ satisfying the condition (5), they can be considered as uncontrolled factors and it is possible to pass over to the following decision making problem:

$$\min_{\vec{x}\in\mathbf{D}} \left\{ \max_{\vec{\lambda}\in\mathbf{D}_\lambda} W(\vec{\lambda}, Q_1(\vec{x}), \ldots, Q_s(\vec{x})) \right\}, \tag{6}$$

where $\mathbf{D}_\lambda$ is the region of feasible values of the weighting coefficients $\vec{\lambda}$, its structure determined by the additional information on the binary relations of preference for the particular optimality criteria.

The weighting coefficients are easily understood as functions from the value of the controlled parameters $\vec{x}$:

$$\lambda_i = \lambda_i(\vec{x}), \qquad i = \overline{1,s}. \tag{7}$$

Thus, considering correlations (7) the initial multiobjective optimization problem (1) is reduced to nonlinear programming problems, the numerical solution of which can be obtained through one of the methods outlined in (Batishchev, 1984).

3 The formalization of the qualitative information on the binary relations of preference for the particular optimality criteria

The structure of the feasible values of the weighting coefficients $\mathbf{D}_\lambda$ is determined by the qualitative information obtained from the decision maker concerning the pairwise comparison of the particular criteria as for their importance. The importance of the criteria is understood here in the sense of either the axiomatic importance theory (Podinovskij, 1979) or the utility theory (Keeney and Raiffa, 1976), which in any case enables the following correlation: if the additional information of the type "the i-th criterion is not less important than the j-th criterion ($Qi \succeq Q_j$)" is obtained, then for weighting coefficients λ_i and λ_j the following interrelation is valid:

$$\lambda_i \geq \lambda_j \qquad \text{when and only when} \qquad Q_i \succeq Q_j \tag{8}$$

Further it will be assumed that in common case the weighting coefficients λ_i, $i = \overline{1,s}$, are normalized with regard to the parameter $R > 0$ and can acquire numerical values not less than a certain nonnegative value λ_0:

$$\mathbf{D}^1_\lambda = \left\{ \vec{\lambda} \;\middle|\; \lambda_i \geq \lambda_0 \geq 0,\ i = \overline{1,s};\ \sum_{i=1}^{s} \lambda_i = R \right\} \tag{9}$$

Suppose that for some L pairs of the particular criteria (not necessarily for all C^2_s feasible pairs) additional information is obtained from the decision maker stating the preference of the i-th criterion to the j-th criterion within the whole set of feasible values of the controlled parameters:

$$\omega_l = \{ Q_i \succeq Q_j \}, \qquad l = \overline{1,L} \tag{10}$$

The qualitative information (10), then, according to (8) and (9) enables decision maker to determine the region of feasible values for the weighting coefficients $\vec{\lambda}$ in the following way:

$$\mathbf{D}^2_\lambda = \left\{ \vec{\lambda} \;\middle|\; \lambda_i \geq \lambda_0 \geq 0,\ i = \overline{1,s};\ \sum_{i=1}^{s} \lambda_i = R;\ \lambda_i \overset{\omega_l}{\geq} \lambda_j,\ l = \overline{1,L} \right\} \tag{11}$$

In a concrete case when the particular criteria are ranked as for their importance ($Q_1 \succeq Q_2 \succeq \cdots \succeq Q_s$), the area $\mathbf{D}_\lambda$ can be written as follows:

$$\mathbf{D}^3_\lambda = \left\{ \vec{\lambda} \;\middle|\; \lambda_i \geq \lambda_0 \geq 0,\ i = \overline{1,s};\ \sum_{i=1}^{s} \lambda_i = R;\ \lambda_i \geq \lambda_{i+1},\ i = \overline{1,s-1} \right\} \tag{12}$$

Let us present the qualitative information (10) as a directed graph of preference $G(X,V)$, where X is a set of vertices, representing particular criteria with V as a set of arcs (edges) connecting the i-th vertex (criterion) with the j-th vertex when and only when the correlation $Q_i \succeq Q_j$ takes place. Let us stream all the vertices of the graph like this: the first level ($q = 1$) will include the vertices having no ingoing arcs; the second level ($q = 2$) will include the vertices having ingoing arcs from the 1st level, etc.; the last level ($q = m \leq s$) will include the vertices which have no outgoing arcs but may accept the arcs going from all the previous levels.

We shall introduce a number of the parameters for each vertex i representing the particular criterion Q_i: I_i as a set of vertices of the graph $G(X, V)$ from which there is a way to the vertex having the number i, the vertex included; n_i is the power of the set I_i.

4 Calculating the weighting coefficients of relative importance through the "guaranteed result" principle

Weighting coefficients $\vec{\lambda}(\vec{x}) \in \mathbf{D}^2_\lambda$, being the optimal solution for the extremal problem

$$\max_{\vec{\lambda} \in \mathbf{D}_\lambda} \Big\{ \sum_{i=1}^{s} \lambda_i Q_i(\vec{x}) \Big\} \tag{13}$$

are calculated by the following formula (Batishchev, 1984; Batishchev et al., 1987):

$$\lambda_i(\vec{x}) = \begin{cases} \dfrac{R - (s - n_r)\lambda_0}{n_r}, & \text{for } i \in I_r; \\ \lambda_0, & \text{for } i \in I \setminus I_r \end{cases} \tag{14}$$

where $I = \{1, 2, \ldots, s\}$; $0 \leq \lambda \leq R/s$; the index r is determined by the condition:

$$q_r = \max_{1 \leq k \leq s} q_k;$$

where

$$q_k = \frac{R - (s - n_k)\lambda_0}{n_k} \sum_{j \in I_k} Q_j(\vec{x}) + \lambda_0 \sum_{i \in I \setminus I_k} Q_i(\vec{x}) \tag{15}$$

It is not difficult to see that if information on the importance of the particular criteria (area $\mathbf{D}^1_\lambda$) is not available the weighting coefficients $\vec{\lambda}(\vec{x}) \in \mathbf{D}^1_\lambda$ equal to the values obtained in reference (Germeier, 1971) and are calculated by the formula:

$$\lambda_i(\vec{x}) = \begin{cases} R - (s-1)\lambda_0, & \text{for } Q_j(\vec{x}) = \max\limits_{1 \leq k \leq s} Q_k(x); \\ \lambda_0, & \text{for all } i = \overline{1, s}, \ i \neq j \end{cases} \tag{16}$$

If the decision maker is able to specify the information on the interrelations between the weighting coefficients λ_i and λ_{i+1}, which follow the binary relation $Q_i \succeq Q_{i+1}$, we obtain the area of feasible values for the weighting coefficients

$$\begin{aligned} \mathbf{D}^4_\lambda \ = \ & \Big\{ \vec{\lambda} \ \Big| \ \lambda_i \geq 0, \ i = 1, s; \ \lambda_i \geq \xi_i \lambda_{i+1} + \varepsilon_i, \ \xi_i > 0, \ \varepsilon_i \geq 0, \ i = \overline{1, s-1}; \\ & \lambda_s \geq \lambda_0 = \xi_s > 0; \ \sum_{i=1}^{s} \lambda_i = R \Big\} \end{aligned} \tag{17}$$

The optimal solution of the extremal problem (13) for the area of feasible values (17) is calculated by means of the following expressions (Anuchin, 1985):

$$\lambda_i(\vec{x}) = \begin{cases} R'\dot{\xi}_i^{r-1} \Big/ \sum_{k=1}^{r} \dot{\xi}_k^{r-1} + \dot{\xi}_i^{s} + \sum_{k=i}^{s-1}(\dot{\xi}_i^{k-1}\varepsilon_k), & i = \overline{1,r} \\ \dot{\xi}_i^{s} + \sum_{k=i}^{s-1}(\dot{\xi}_i^{k-1}\varepsilon_k), & i = \overline{r+1,s}; \end{cases} \tag{18}$$

where

$$R' = R - \sum_{k=1}^{s}\Big(\dot{\xi}_k^{s} + \sum_{i=k}^{s-1}\varepsilon_i\dot{\xi}_k^{i-1}\Big) \geq 0; \tag{19}$$

$$\dot{\xi}_l^{k} = \begin{cases} \prod_{i=l}^{k}\xi_i, & l \leq k \\ 1, & l > k. \end{cases} \tag{20}$$

The index r is determined by the condition:

$$q_r = \max_{1 \leq k \leq s} q_k;$$

where

$$q_k = \sum_{i-1}^{k}\dot{\xi}_i^{k-1}Q_i(\vec{x}) \Big/ \sum_{i=1}^{r}\dot{\xi}_i^{r-1}. \tag{21}$$

The weighting coefficients $\vec{\lambda}(\vec{x}) \in \mathbf{D}_\lambda$, being the optimal solution for the extremal problem

$$\max_{\lambda \in \mathbf{D}_\lambda} \max_{1 \leq i \leq s} (\lambda_i Q_i(\vec{x})) \tag{22}$$

are calculated by the formula (14)–(15), where the function q_k needed to determine the index r is obtained by the formula:

$$q_k = \frac{R - (s - n_k)\lambda_o}{n_k} Q_k(\vec{x}). \tag{23}$$

References

Anuchin, V.F. (1985). Postroenie individual'nykh koehffitsientov vazhnocti na osnovie utochnennoj kachestvennoj informacii o vazhnocti kriteriov. Modelirovanie i optimizaciya slozhnykh sistem. Voronezh, izd. Voronezhskovo politekhnicheskogo instituta, (in Russian).

Batishchev, D.I. (1984). Optimal design methods. Moscow, Radio i sviaz, 248p., (in Russian).

Batishchev, D.I. and Anuchin, V.F. and Shaposhnikov, D.E. (1987). Weighting coefficients determination from the qualitative information on the importance of particular criteria of optimality. Internationale Tagung: Mathematische Optimerung — Theorie und Anwendungen, Eisenach, 16–20 November 1987. Technische Hochschule Ilmenau, DDR, pp. 202–205.

Cohon, J.L. (1978). Multiobjective Programming and Planning. *Mathematics in science and engneering*, Vol. 140, Academic Press, New York, 333p.

Germeier, Ju.B. (1971). Introduction into the theory of operation research. Moscow, Nauka, 383p., (in Russian).

Keeney, R.I. and Raiffa, H. (1976). Decisions with multiple objectives: preferences and value trade-offs. Wiley, New York.

Lewandowski, A. and Wierzbicki, A. (1988). Aspiration Based Decision Analysis and Support. Part 1: Theoretical and Methodological Backgrounds. Working Paper WP-88-03, IIASA, Laxenburg, Austria.

Mikhalevich, V.S. and Volkovich, V.L. (1982). Computational methods in study and design of complex systems. Moscow, Nauka, 286p., (in Russian).

Podinovskij, V.V. (1979). An Axiomatic Solution of the Problem of Criteria Importance in Multicriterial Problems. In N.N. Moiseev, ed., Contemporary State of Operation Research Theory. Moscow, Nauka, pp. 117–145, (in Russian).

Zionts, S., ed. (1978). Multiple Criteria Problem Solving. *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag, Berlin, 567p.

Multiple Objective Decision-Making Aid Using Analysis of Inconsistency Between Constraints

A. B. Bordetsky, V. A. Tsygankov
Chelyabinsk Polytechnical Institute
Chelyabinsk, USSR

1 Introduction

Since a real decision problems in the initial stages of the computer-aided design, urban and resource planning and other applications involves hardly inconsistent and ill-defined constraints a practically useful multi-criteria decision making (MCDM) system must include some extensions for preceding overview of feasible and roughly efficient solutions.

The report deals with the variants of such decision-making techniques which employ a sequential deletion of inconsistency between constraints via a progressive articulation of preference and a feature of the corresponding decision-support system (DSS). Each step of man-machine dialog with such DSS results in achievement and extension of some collectively satisfied subsystem of constraints with reference to the form of feasible set and a combination of constraints. As a general information computer gives various specifications of collective compatibility of constraints: committee of inequalities, committee of solutions etc. Then an auxiliary information is characteristics of achieved feasible subset: visual crossections, clusters of feasible solutions, "center" of feasible solutions, "center" of feasible subset etc, recommendations on relaxation levels of unsatisfied constraints. Decision-maker (DM) varies weights of objectives, upper and lower bounds of unsatisfied constraints. According to composition of general and auxiliary information one receives ways of interaction which differ by effort of decision-maker. Process ends while DM satisfies with non-empty feasible set and configuration of roughly efficient set. Then a STEM-rule algorithm can be used as a final step of dialog.

The main features of DSS are as follows:

1. Direct and iterative solvers of inequalities combined in a problem solving mechanism;

2. Choice rules for different ways of dialog in resolving inconsistent situations between criterial and direct constraints together with expert-strategy rules combined in knowledge base;

3. Spreadsheet user-friendly interface to different MCDM models and realisation of choice rules in form of special recalculation menu and options with help of original spreadsheet's tool kit.

In such environment reasoning ability of DM about changing upper and lower bounds of criteria are combined with result of numerical experiments on constraints in united knowledge base. With help of special choice rules we can also add various optimization capabilities to DSS within interpretation to equivalent systems of inequalities. Using such MCDM aid we can build special types of semi-expert DSS, according to Keen's classification (Keen, 1987) or types of expert systems with quantitive knowledge base according to (Konopasek and Jaraman, 1984).

2 Decision making process

Basic objective MCDM model DM specifies as a system of inconsistency constraints:

$$\begin{array}{l} \theta_i : \ a_i \leq Y_i \leq b_i, \quad \{i\}_1^m = I, \quad \bar{x} \in X \subset E^4 \\ Y_i = f_i(\bar{x}), \ \ f_i(\bar{x}) \in \{E^4 \to E^1\} \ - \ \text{functional} \end{array} \tag{2.1}$$

form of relationship between decision criterion and alternative, where:

1. $M = \bigcap_{i \in I} M_i = \emptyset$, $M_i = \{\bar{x} \in X \mid f_i(\bar{x}) \in [a_i, b_i]\}$;
2. constraints $\{\theta_i\}$ are not equal by priority to DM;
3. model attributes $\bar{d} = \{a_1, \ldots, a_m; b_1, \ldots, b_m\}$ are variables for DM.

Group and individual inconsistents between constraints (2.1) reflect various conflicts in terms of criterial set.

The main man-machine iterative decision process is as follows:

$$\left\{ I^t_{com}(M^t) \stackrel{CHR_i}{\longrightarrow} I^t_{DM}(var\ \bar{d}) \stackrel{CHR_i}{\longrightarrow} M_{t+1} \right\}_1^N \tag{2.2}$$

where

I^t_{com} computer information about inconsistency of system (2.1);

t step of interaction;

I^t_{DM} information from DM about new values $\{\bar{d}^t\}$ (model of DM reaction on I^t_{com});

CHR_i choice rule (Han-Lin Li, 1987) between different mathematical models of inconsistency I^t_{com} and types of human reactions I^t_{DM}.

Reffering to (Larichev, 1987) we detalise I^t_{com} as

$$I^t_{com} - \{I^t_{BAS}(I), I^t_{add}(\overline{x})\},$$

I^t_{BAS} basic computer information , which reflects inconsistency in terms of indexes $\{I\}$

I^t_{add} additional uses-friendly information about achieved non empty subset in terms of alternatives $\{\overline{x}\}$.

Different combinations of $(I^t_{BAS}, I^t_{add}, I^t_{DM})$, being united in corresponding choice-rules are saved as a part of DSS knowledge base (fig. 1).

The DM can select any CHR_i, which differ in type of criterial functions $\{f_i(\bar{x})\}$ (linear, non-linear, table-like, etc.), psychological correctness (Larichev, 1987) and other features. As a result DM can achieve different feasible subset and different parts of efficient solution set. In such a way DM receives an opportunity to learn capabilities of MCDM problem within man-machine process (2.2).

3 Examples of choice rules

Some MCDM algorithms give an example of I_{com}, I_{add}, I_{DM} and variants of choice rules.

3.1 Choice rule based on (Glushkov et al., 1983)

I^t_{BAS} : $\bar{x}^* \in \{T_i\}$; T_i – table approximation of criteria; enumeration of constraints non satisfied in point

I^t_{add} : no

I_{DM} : $\{a_i(\bar{x}^*),\ b_i(\bar{x}^*),$ another $\bar{x}^*\}$

3.2 Choice rule based on (Sobol and Setnikov, 1981)

I^t_{BAS} : $\bar{x}^* \in M_i$, i – number of more essential criterial constraint with respect to direct constraints $\{e_j \leq x_j \leq d_j\}$, $j = \overline{1,m}$

I^t_{add} : recommendation about selection of new bounds for and coefficients $\prod[e_j, d_j]$ and coefficients $\{\alpha_{ij}\}$ of linear constraints $f_j = \sum_{j=1}^m \alpha_{ij} x_j$ nonsatisfied at $\bar{x}^*$

I^t_{DM} : $\{a_i, b_i, e_j, d_j\}$

3.3 Choice rule based on (Bushenkov and Lotov, 1987)

I^t_{BAS} : no

I^t_{add} : visual projections of achievement set for linear functions in $a_i \leq f_i(\bar{x}) \leq b_i$

I^t_{DM} : $\{\bar{x}^*, a_i, b_i\}$

3.4 Choice rule based on (Jeremin, Mazurov, 1979)

I^t_{BAS} : committee set $\{(x_1, \ldots, x_g)\} \in E^4 \mid |I_k(\bar{x})| \geq \frac{1}{2}|I|\}$, where $I_k(\bar{x}) = \{i \mid f_i(\bar{x}) \in [a_i, b_i]\}$

I^t_{add} : no

I^t_{DM} : no

3.5 Choice rule based on (Bordetsky, 1988)

I^t_{BAS} : committee of inequalities

$$I_k = \left\{ i \in I_g \subseteq I \mid \max_{\{I_g\}} \sum_{i \in I_g} c_i \right\}, \quad c_i \; - \text{ cardinal weighting coefficient}$$

$$I_g = \left\{ i \in I \mid \bigcap_{i \in I_g} M_i \neq \emptyset \; \& \bigcap_{i \in I'_g} M_i = \emptyset, I'_g = I_g \cup (i \in I \setminus I_k) \right\} \; -$$

non plus subsystem from I.

I^t_{add} : table approximation and visual crossections of committee feasible set, recommendations about selection of new bounds for nonsatisfied constraints

I^t_{DM} : $\{a, b_i, e_j, d_j\}$.

3.6 Choice rule based on (Tõrn, 1980)

I^t_{BAS} : no

I^t_{add} : clusters of feasible points $\{\bar{x}_1, \ldots, \bar{x}_k\} \in \{(\bigcap_{i \in I} M_i \neq \emptyset)\}$

I^t_{DM} : no

Examples show that for linear MCDM problems we have more possibilities to combine different CHR, but for nonlinear constraints practically useful there are choice rules (3.1) and partly (only for convex functions (3.5)).

4 Expert rules

Individual strategy of DM we can represent two types of expert rules:

1. production rule in form of condition-action pairs, where action means direct call any of the basic constraints solvers,

2. functional rules in form of special linear constraints.

An interesting feature of our system is that within basic constraints solvers we can support not only main MCDM process (1.2) but also represent subjective decision rules in the same constraints like form with help of committee choice rule (3.5).

Let $\bar{d}^t_{sat}$: $\bar{d}^t \xrightarrow{CHR_i} \bar{d}^{t+1}$ — satisfied variations of bounds; $\bar{d}^t_{uns}$ — unsatisfied, but not extracted by DM.

First we construct special system of constraints

$$\begin{cases} \sum_{i=1}^{m} \alpha_i d^t_{isat} & \geq \; 0 \\ \sum_{i=1}^{m} \alpha_i d^t_{iuns} & \leq \; 0 \end{cases} \tag{4.1}$$

Then we call main MCDM process (2.2) with committee choice-rule (3.5) and select one decision from committee feasible set

$$\bar{\alpha}^* = (\alpha^*_1, \ldots, \alpha^*_m) \in \left(\bigcap_{i \in I_k} M_i \neq \emptyset \right)$$

as a decision rule coefficients vector

$$\sum_{i=1}^{m} \alpha_i^* d_i \geq 0.$$

5 Basic constraint's solvers

These algorithms are realised as system's primitives. Solvers are lower level programs in call-structure of choice rules as a components of DSS. Types of solvers are as follows.

1. Direct solver for estimation of feasible solutions of non-linear inequalities within straight replacement of variables.

2. Iterative solvers for linear constraints include iterative solver based on the sequential projections algorithm:

$$\begin{aligned} x_j^{k+1} &= x_j^k + h \cdot C_{ij}, \quad j = \overline{1,n} \\ \text{where} \quad h &= \left(\begin{Bmatrix} a_i \\ b_i \end{Bmatrix} - f(\bar{x}^k)\begin{Bmatrix} + \\ - \end{Bmatrix}(b_i - a_i) \cdot P_k\right) / \sum_{j=1}^{n} C_{ij}^2 \end{aligned} \tag{5.1}$$

For nonlinear constraints solver mechanism contains iterative solver based on random LP_γ (Sobol, Statnikov, 1981) sequences:

$$x_j^{k+1} = x_j^k(LP_\gamma - \text{operator}) \tag{5.2}$$

6 Software characteristics

The main purpose of MCDM technique based on the inconsistency analysis between constraints is to realise various "what-if" interactions for learning the structure of feasible set and capabilities of MCDM objective model. One of the best friendly interfaces for "what-if" analysis is a speadsheet interface. The kernel of DSS is instrumental spreadsheet. Problem-oriented packages are formed as static combination of sheets produced by spreadsheet's edit tool kit. Recalculation menu is open for additions. Choice rules are realised as a special recalculation menu-driver commands in two forms. In "hard" form choice rule is realised directly within "call-structure" of recalculation tree. In "soft" form recalculation menu is free for choice, but choice rules are realised as a special prompts in the help-graph.

References

Bordetsky, A. B. (1988). Interactive System for Searching and Selection of Project Feasible Solutions within Computer Unitation. Questions of Cybernetics, Modelling Processes in Man-Machine Systems. Moscow, pp. 16–27 (in Russian).

Bushenkov, V. A., Lotov, A. V. (1987). Methods on Constructing and Using Holistic Achievement Sets. Moscow Computer Center Academy of Science.

Glushkov, V. M., Michalevich, V. S., Volkovich, V. L., Dolenko, G. A. (1983). System Optimisation in Multicriteria Linear Problems within interval representation of preferences. *Cybernetics* 3, pp. 1–87.

Han-Lin, Li. (1987). Solving Discrete Multicriteria Decision Problems Based on Logic Based Decision Support Systems. *Decision Support Systems* 3, pp. 101–119.

Jeremin, I. I., Masurov V. D. (1979). Nonstatic Processes in Mathematical Programming. Moscow Science, (in Russian).

Keen, P. C. W. (1987). Decision Support Systems : The Next Decade. *Decision Support Systems* 3, pp. 253–265.

Konopasek, M., Jaramen, S. (1984). Expert Systems for Personal Computers. The TK! Solver Approach. *Byte* 5, pp. 139–154.

Larichev, O. I. (1987). Objective Models and Subjective Decisions. Moscow Science, (in Russian).

Sobol, I. M., Statnikov, R. B. (1981). Selection of Optimal Solutions in Multicriteria Problems. Moscow Science, (in Russian).

Tõrn, A. A. (1980). A Sampling-Search-Clustering Approach for Exploring the Feasible/Efficient Solutions of MCDM. *Computers and Ops.Res.*, vol.7, pp. 67–79.

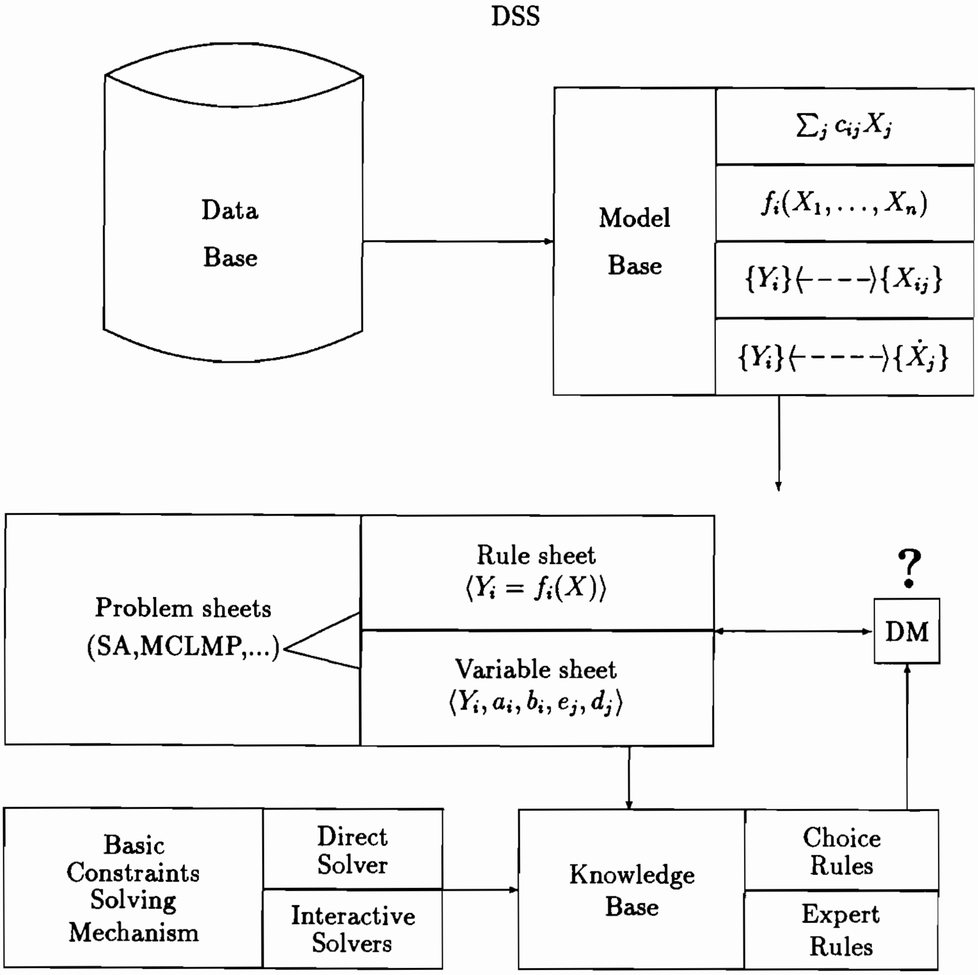

DSS
Data
Base
Model
Base
$\sum_j c_{ij} X_j$
$f_i(X_1, \ldots, X_n)$
$\{Y_i\} \langle\!-\!-\!-\!\rangle \{X_{ij}\}$
$\{Y_i\} \langle\!-\!-\!-\!-\!\rangle \{\dot{X}_j\}$
Problem sheets
(SA,MCLMP,...)
Rule sheet
$\langle Y_i = f_i(X) \rangle$
Variable sheet
$\langle Y_i, a_i, b_i, e_j, d_j \rangle$
?
DM
Basic
Constraints
Solving
Mechanism
Direct
Solver
Interactive
Solvers
Knowledge
Base
Choice
Rules
Expert
Rules

Figure 1

Multicriteria Problems with Objective Models

Oleg I. Larichev
Institute for Systems Studies
USSR Academy of Sciences
Moscow, USSR

Introduction

The multicriteria mathematical programming problems to be considered at the conference, are quite specific differing from both operations research problems and many multiattribute decision problems.

In multicriteria mathematical programming problems, like in many operations research problems, there is a reliable (objective) model of the object under study, i.e. a set of perfectly verified relationships between the basic object variables. However in contrast to operations research, there is a variety of requirements to the quality of solution, i.e. multiplecriteria.

The latter constitutes a specific characteristic of the widespread real-life problems. The choice of the best decision alternative places demand for a tradeoff between the estimates against different criteria. The problem conditions lack information permitting a tradeoff. Hence, it cannot be found through objective calculations.

The analysis of many real-life problems, the operations researchers had dealt with, has naturally produced a class of multicriteria problems lacking information which makes it possible to find the best decision.

Since a decision must, somehow or other, be made, the shortage of information required for the best alternative choice, should be eliminated. This can be done only by people on the basis of experience and intuition.

The evolvement of preferences and human policy as an inseparable part of the problem drastically changes both its essence and solution techniques. There arises a plethora of questions characteristic of all decision problems:

1. How to help man validate the rationality of his decision?
2. How to elicit information from man in the process of problem solution?
3. How to verify the consistency of information elicited from man?
4. How to help man analyze the opportunities for a tradeoff between the criteria, determined by the objective model of the considered problem?

The four questions, we believe, are fundamental for constructing man-machine solution procedures of multicriteria problems with objective models.

Hence we infer from here that transition from single-

Now we turn to the search for answers to the above questions using a multicriteria linear programming problem (MLPP) as an example:

Find vector $x = (x_1, \ldots, x_n)$ belonging to domain

$$D = \{ Ax = b;\ x_i \geq 0,\ i = 1, \ldots, n \}$$

where A is $p \times n$-matrix; b is p-vector maximizing (or minimizing) the set of objective functions

$$c_k(x) = \sum_{i=1}^{n} c_{ik} x_i; \qquad k = 1, \ldots, N$$

for the most preferable ratio between their values in decision point. This requirement means: in a variety of X effective (Pareto-optimal) decisions one should seek for X^* decisions corresponding to the extremum of a priori unknown decision maker's utility function.

Analysis shows that there are three groups of methods developed by different authors for solving MLPP. According to one of them, at the analysis phase decision maker compares changes in the estimates of a pair of criteria and/or assigns a satisfactory value against one criterion. This idea was first advanced in STEM procedure (Benayon et al., 1971). According to the second idea, decision maker specifies direction in the criterion space along which his implicit utility function increases (analogy of gradient method). The most familiar procedure of this type is the one of Dyer-Gioffrion (Dyer, 1976). The third version of man-machine procedure construction boils down to gradual localization of ε-vicinity of optimal point, and is related with truncation of feasible decision domain.

Validation of decision rationality

Though solution of multicriteria problems with objective models depends on the decision maker preferences, this does not imply that he "makes whatever he likes". An individual must be rational in business decisions so that to be able to convince others, explain to them the motives of his choice, the logics of his subjective model. Any decision maker preferences should, therefore, be within the frameworks of some rational system. Very often his policy, his subjective model is, in fact, manifestation of the policy of a group of persons surrounding him. This does not make the model more objective, rather it becomes as if more stable — it remains the same for any decision maker from some group possessing a common preference, a common "world outlook". Often this unity is largely determined by the status of the organization, the given group of managers belongs to, its environments.

This forces the decision maker to explain the derived decision. Hence, on arriving to a decision, he has first to trace the logics of his successive decisions for himself and only then explain this logics to others.

Different methods of MLPP solution provide the decision maker with different opportunities for explaining the choice. Thus, methods involving assignment of the so-called "ideal" decisions (procedures by M. Zeleny, 1976; A. Wierzbicki, 1980) provide a good opportunity for explanations in the form of an "ideal" points trajectory. Transition from one "ideal" point to the other can be explained by a real decision obtained on the margin of a feasible domain. As acceptable for explanation are the methods of alternate assignment of satisfactory criteria values (Spronk method, Nijkamp and Spronk, 1980; STEM, Benayon et al., 1971). Explanation can be in the form of a set of curves of tradeoffs between pairs of criteria.

Much less suitable for explanation are methods associated with the computation of utility function gradients (e.g. Dyer-Gioffrion procedures, Dyer, 1976). This procedure necessitates a search for local gradients coefficients in different points. It is very difficult to characterize the logic of change in the direction of search and points of decisions along these directions in the decision space.

Admissible information processing operations

The majority of data processing operations, exercised by decision makers in man-machine procedures, can be classified in three groups: operations with names of criteria, operations with separate criteria estimates of one alternative, operations with alternatives presented as a set of criteria estimates. We shall refer to an operation as elementary if it cannot be partitioned to a larger number of operations relating to objects of the same group.

Elementary operations can be grouped in the following classes (Larichev et al., 1987; Larichev and Nikiforov, 1986):

(a) complex (C) if psychological research indicates that in performing these operations the decision maker is often inconsistent and/or makes use of simplifying strategies (e.g. eliminates a part of criteria);

(b) complex, except for small dimension problems (CS) if psychological research shows that the decision maker successfully performs these operations on small problems (2–3 criteria, 2–3 alternatives), but on larger problems he is often inconsistent and/or employs simplifying strategies;

(c) admissible (A) if the research indicates that the decision maker can manage them reliably , i.e. with a small number of inconsistencies, and using complex strategies (e.g. combination of several criteria estimates);

(d) uncertain (U, UC, UA) if an insufficient number of studies on these operations have been conducted but it is possible to judge about them by analogy (UC, UA).

The analysis of different man-machine procedures helped distinguish a small number (about 10) of elementary operations (see Overview, Larichev et al., 1987). A thorough examination of psychological literature made it possible to distinguish a cluster of procedures using only correct elementary operations of information elicitation from decision makers. All of them relate to a class of search for satisfactory criteria values (IMGP (Nijkamp and Spronk, 1980), STEM (Benayon et al., 1971), etc.).

Hence, man-machine procedures using a search for a pairwise tradeoff between criteria are more correct in terms of information elicitation from decision makers.

Human errors in search process

The use of correct operations of information elicitation from decision makers essentially reduces chances of errors or employment of simplified strategies by decision makers. It is, however, impossible to completely rule out human errors, for they can be brought about not only by cognitive constraints but also carelessness or fatigue. Also, errors can emerge at a time of learning when decision maker has not yet arrived at a compromise between criteria following examination of feasible values domain. Accordingly, it is necessary to secure a low sensitivity of man-machine procedures to decision maker and expert errors. A good means for reducing that of experts, estimating alternatives against many criteria (given a discrete variety of alternatives) is an interval assessment method first suggested by R. Steuer (Steuer and Schuler, 1978).

Procedures where a random error does not eliminate the feasible values domain from consideration are known to have a reduced sensitivity to decision maker errors. We used this criterion (Larichev, 1987) in comparing several man-machine procedures. Six out of 19 considered procedures did not meet this criterion.

It should be noted that all procedures give inadequate attention to possible decision maker errors. The methods of decision maker check for consistency used in a number of decision methods with subjective models (Gnedenko et al., 1979) boil down to duplication (directly or indirectly) of information elicited from decision maker.

Decision maker learning in the process of search

Everybody who has ever employed man-machine procedures for solving multicriteria mathematical programming problems knows that at the early steps of the procedure decision maker wants "everything at once", i.e. is willing to reach extremum against all criteria at a time. Only after familiarizing himself with the domain of feasible solutions he comes to understand the impossibility of this, and starts developing a more realistic approach.

All man-machine procedures, to some or other extent, provide opportunities for decision maker learning. Some of them, however, are better than others. Assuringly, at a time of learning decision maker should rather explore capacities of the extreme criteria values by reviewing criteria in turn rather than concurrently. This opportunity is provided by the methods of search for satisfactory criteria values.

Directions of further search

The procedures of MLPP solution, accounting for the specifics of different practical problems, have a long history. The recent overviews comparing man-machine procedures (Wallenius, 1975; Larichev and Polyakov, 1980; Polishchuk and Mirkin, 1980; Larichev and Nikiforov, 1986) reflect a desire to develop MLPP solution techniques with regard to the necessary criteria of their quality.

The existing man-machine procedures of MLPP solution get increasingly sophisticated. Still they are capable of advancing further. It is necessary to improve three basic components of man-machine procedures:

1. Methods of information elicitation from man with regard to specifics and limitations of human information processing system.

   The advances in solving this problem will make it possible to scientifically validate man-machine procedure in terms of psychological, mathematical, and informatics criteria.

2. Organization of an effective man-machine interface.

   There is a need for analysis of different types of information presentation. A special attention should be given to graphical images — cross-section of multicriteria space (Lotov, 1972), trajectory of "a Pareto race" (Korhonen and Wallenius, 1986).

3. Methods of effective solution of mathematical programming problems.

   The methods of multicriteria mathematical programming problem solution are based on iterative solution of single criterion problems. They can be rather complex: discrete, discrete-continuous, integer, etc. There is a need for methods of rapid solution of these problems with an acceptable accuracy (which is highly important for NP-complex problems). Otherwise we shall fail to maintain man-machine interaction.

References

Benayon, R., Montgolfier, J., Tergny, J., Larichev, O. (1971). Linear Programming with Multiple Objective Functions: STEP Method (STEM). *Math. Program.*, Vol. 1, No. 3.

Dyer, J. (1976). Multiobjective programming with man-machine procedures. In: Issues of analysis and decision procedures. Moscow, Mir Publishers, 1976.

Gnedenko, L.S., Larichev, O.I., Moshkovich, H.M., Furems, E.M. (1979). Procedure of quasiordering a set of multicriteria alternatives on the basis of credible information about decision maker preferences. *Avtomatika i telemehanika*, No. 9.

Korhonen, P. and Wallenius, J. (1986). A Pareto race. Working Paper of Arizona State University, DIS 85/86–13.

Larichev, O.I. (1987). Objective models and subjective decisions. Moscow, Nauka Publishers.

Larichev, O.I. and Nikiforov, A.D. (1986). Analytical overview of solution procedures for multicriteria mathematical programming problems. *Ekonomika i mat. metody*, Vol. 22, No. 3.

Larichev, O.I. and Polyakov, O.A. (1980). Man-machine procedures of solving multicriteria mathematical programming problems. *Ekonomika i mat. metody*, Vol. 16, No. 1.

Larichev, O., Polyakov, O. and Nikiforov, A. (1987). Multicriteria linear programming problems. *Journal of Economic Psychology*, No. 8.

Lotov, A.B. (1972). A numerical method of constructing ?sets of attainability? in linear systems. *Zhurnal vychisl. matem. i mat. fiziki*, Vol. 12, No. 3.

Michalevich, V.S., Volkovich, V.L. et al. (1989). Multicriteria mathematical programming problems, the methods of solution and software, in this volume.

Nijkamp, P. and Spronk, J. (1980). Interactive multiple goal programming: an evaluation and some results. In G. Fandel and T. Gal, eds., Multiple criteria decision making. Berlin, Springer Verlag, Vol. 177.

Polishchuk, L.I. and Mirkin, B.G. (1980). Multicriteria problems of econometric modelling and their solution techniques. In: Data analysis and decision models. Novosibirsk, Institute of Economics and Industrial Engineering of the Siberian Division of the USSR Academy of Sciences.

Steuer, R. and Schuler, A. (1978). An interactive multiple objective linear programming approach to a problem in forest management. *Oper. Res.*, Vol. 26, No. 2.

Wallenius, J. (1975). Comparable evaluation of some interactive approaches to multicriterion optimization. *Management Science*, Vol. 21, No. 12.

Wierzbicki, A.P. (1980). The use of reference objectives in multiobjective optimization. In G. Fandel and T. Gal, eds., Multiple criteria decision making. Berlin, Springer Verlag, Vol. 177.

Zeleny, M. (1976). The theory of displaced ideal. In H. Thitiez and S. Zionts, eds., Multiple criteria decision making. Berlin, Springer Verlag, Vol. 130.

Soviet Multi-Objective Programming Methods: An Overview

Elliot R. Lieberman
Johns Hopkins University, Baltimore, USA *

Abstract

Both Western and Soviet surveys of multiple criteria decision making (MCDM) have given virtually no attention to the sizable, important, and original body of Soviet research in multi-objective programming (MOP). To begin to correct this situation, this paper identifies and classifies some of the most noteworthy Soviet MOP research. Using a classification scheme similar to that found in Hwang and Masud (1979) and Evans (1984), we group methods based on when they elicit preference information from the decision maker and what form this information takes.

1 No Articulation of Preferences

The first category of methods are those which do not require preference information from the decision maker — neither before, during, nor after the solution process. Underlying these methods is the often unstated assumption that a multi-objective problem has an "optimal" solution, which can be found by transforming the vector problem into a corresponding scalar problem.

1.1 Ideal Distance Minimization Method (Salukvadze 1971 a,b; 1979)

This method is historically significant, being the first attempt, either in the East or West, to employ the now widely used concept of an "ideal point" to scalarize problems having multiple objectives. Devised to solve dynamic control problems with multiple (vector) functionals, the method minimizes the Euclidean distance between the ideal trajectory $I^*(\bar{u})$ and the set of feasible trajectories $I(\bar{u})$:

$$\inf_{\bar{u}(t)} R(\bar{u}) = \|I(\bar{u}) - I^*(\bar{u})\| = \sum_{\alpha=1}^{k} \left[I_\alpha(\bar{u}) - I_\alpha(\bar{u}^{(\alpha)})\right]^2$$

*Research for this article was supported in part by a grant from the International Research and Exchanges Board (IREX), with funds provided by the National Endowment for the Humanities, the Andrew W. Mellon Foundation, and the Department of State. None of these organizations is responsible for the views expressed.

1.2 Maximal Effectiveness Principle (Khomenyuk 1977a,b; 1983)

Instead of using the Euclidean distance metric to scalarize the multi-objective programming problem, this method assumes that all objectives are being maximized and seeks the solutions which maximizes the minimum relative attainment, $\lambda_i(\bar{x})$, by any objective, $Z_i(\bar{x})$, of its ideal reference value, $Z_i^*(\bar{x})$.

$$\lambda_i(\bar{x}) = \frac{Z_i(\bar{x}) - Z_i^{worst}(\bar{x})}{Z_i^*(\bar{x}) - Z_i^{worst}(\bar{x})}, \quad i = 1, \ldots, p$$

The method's developers claim that it is one of the few multi-objective programming procedures that can be "axiomatically substantiated."

1.3 Velichenko's Minimax Method (Velichenko 1975)

Intended for optimal control problems where multiple objectives are being minimized, this method simply proposes seeking a minimax solution — i.e., a trajectory which gets the best possible value from the worst performing objective function Φ_ω.

$$\min_{\bar{x}(T)} \max_{\omega \in \Omega} \{\Phi_\omega [\bar{x}(T), T]\}$$

2 A Priori Articulation of Preferences

While only one Soviet method falls among those requiring the a priori involvement of a decision maker, the particular method involves quite an unconventional use of multi-objective procedures.

2.1 Multi-Objective Decomposition (Krasnoshchekov, Morozov, Fedorov 1979a,b,c)

The most common approach to solving multi-objective programming problems has been to transform them into more tractable scalar problems. This method does the reverse: complex single objective problems are converted into multi-objective problems in an attempt to reduce the size of the initial set of candidate solutions. This represents an a priori approach because the original problem's scalar function is most often defined ahead of time through informal consultations with a decision maker.

3 Interactive Articulation of Preferences

The most active area of Soviet multi-objective programming research is in interactive methods — those procedures where phases of computation alternate with phases of decision making. The methods in this category are grouped according to the type of information supplied by the decision maker. **Methods Using Target Values**

3.1 Hierarchical/Decomposition (Korotkova 1978, 1982, 1983)

In this method complex multi-objective problems are conceptualized as hierarchical, multi-level systems whose solutions can be found by solving a series of computationally manageable subproblems associated with the various subsystems in the hierarchy. These subproblems are created by successively selecting a different objective from the original multi-objective problem as a subproblem's scalar objective function while the remaining $n-1$ objectives are converted into that subproblem's constraints. Then, the decision maker sets and readjusts target objective function values, and the resulting subsystem problems are iteratively solved until an equilibrium solution is derived which achieves all the subsystems' assigned target values.

3.2 STEM Method (Benayoun, de Montgolfier, Tergny, Larichev 1971; Benayoun, Larichev, de Montgolfier, Tergny 1971)

The product of French and Soviet collaboration, the STEM method is said to be the first interactive multi-objective method applied in practice. Despite its often noted limitations (Steuer 1986:365; Cohon 1978:203), this method appears to enjoy wide use at the Institute for Systems Study, one of Moscow's leading research centers. Being well known both in the West and the USSR, the method's specifics will not be described here.

3.3 Multi-Objective Graph Theory (Dubov, Shmul'yan 1973)

In one of the earliest attempts to introduce multiple objectives into graph theory, the STEM method was adapted to finding the shortest non-inferior path in a directed graph Γ, having n+1 nodes and two objectives, L_1, L_2. (Figure 1) The procedure was used in

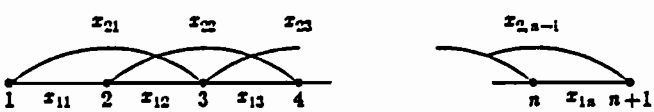


Figure 1: Shortest path problem: graph Γ.

designing building facades consisting of two types of standard prefabricated modules.

3.4 Method of Constraints (Mikhalevich, Volkovich 1982)

An extremely versatile approach applicable to problems regardless of the functional form of their objectives or constraints, this method cleverly transforms multi-objective problems so that their solution simply becomes a matter of checking the consistency of a system of constraints. The method can be applied to a wide variety of difficult-to-solve problems — non-linear, discrete, and hierarchical — where other methods would prove impractical or completely unusable.

3.5 LP_τ Method (Sobol', Statnikov 1981; 1982)

Making use of advanced sampling techniques based on so-called LP_τ sequences, this method is not only applicable to otherwise insolvable problems having perversely behaved feasible regions, but it is also particularly adept at familiarizing a decision maker with the full range of decision alternatives and objective function values — a feature which recent studies have found to be particularly desirable in interactive methods.

Methods Ranking Alternatives or Objectives

3.6 Random Search Method (Bedel'baev, Dubov, Shmul'yan 1976; Popkov, Shmul'yan, Ikoeva, Kabakov 1974)

In this method the decision maker's ranking of previously generated non-inferior solutions $\bar{x}^{(s-1)}$ and $\bar{x}^{(s)}$ is used to direct a random search which generates new parameter values $\bar{\alpha}^{(s+1)}$ for a global function used to scalarize and solve the original multi-objective programming problem

$$\max \bar{L} = \max_{\bar{x} \in X} [L_1(\bar{x}), \ldots, L_m(\bar{x})]$$

3.7 Vector-Relaxation Method (Eiduk 1981, 1983; Rastrigin, Eiduk 1985)

This method applies a gradient search approach to non-linear multi-objective problems. Unlike the single objective case where the gradient search produces a unique search direction, here there are a range of possible search directions and a multiplicity of alternative non-inferior solutions. To address this situation the method imposes additional maximin conditions on the gradient search and involves the decision maker in the process of moving from one candidate solution to another.

3.8 Interactive ϵ-Grid Method (Merkur'ev, Moldavskii 1979; Moldavskii 1980)

This method narrows in on a solution to continuous non-linear multi-objective programming problems by constructing successively tighter sampling grids on the set of non-inferior solutions. Using some parameterized global objective function $\varphi(\bar{\lambda}, F(\bar{x}))$ to aggregate the problem's multiple objectives $(f_1(\bar{x}), \ldots, f_m(\bar{x}))$, the method repeatedly solves for values of parameter $\bar{\lambda}$ falling on ever tighter ϵ-grids of the parameter space.

3.9 Method of Local Improvements (Krasnenker 1975; Kaplinskii, Krasnenker 1975, 1977)

In this method the multi-objective programming problem is viewed simply as a variation of the stochastic programming problem, where the uncertainty, depicted by some random variables $\bar{Y}$, concerns the relative importance of the various objectives. At each iteration a gradient-type descent algorithm, developed by the method's creators for general stochastic programming problems, is used to revise the probability distributions for these random variables.

3.10 Pareto Boundary Maps (Polishchuk 1978, 1980, 1981; Polishchuk, Mirkin 1980)

Rather than a single method, Pareto Boundary Maps refer to a whole class of methods which first aggregate a problem's multiple objective functions into a single objective and then systematically vary the parameters of scalarization while optimizing the resulting functions in order to generate a series of candidate solutions. The developers of this concept examine four commonly used parameterizations (constraint, weighting, bill of goods, and goal programming) as well as present their own which combines parametric scalarization with a gradient search. In all of these methods the interactive component arises from their need to vary the scalarization parameters in response to the decision maker's preferences.

4 A Posteriori Articulation of Preferences

Methods in this category involve the decision maker only after the solution process has been completed. These methods are concerned with generating solution alternatives, not with processing preference information.

4.1 Dynamic Multi-Objective Programming (Dubov 1977, 1978, 1979)

Dividing dynamic programming problems into four categories, Yu. A. Dubov establishes conditions for the existence of non-inferior solutions in the multi-objective variant of such problems. Using these conditions, Dubov outlines an algorithm for determining non-inferior solutions in the simplest case and indicates the features of analogous algorithms for the more complex situations.

4.2 Generalized Reachable Set Method (Lotov 1972, 1973; Bushenkov, Lotov 1980, 1982)

In a truly groundbreaking departure from the methods commonly used to solve linear multi-objective programming problems, the GRS method generates solutions to such problems by making orthogonal projections of the convex polyhedron defined by all the objective functions and constraints onto the subspace of the objective function values. The result G_z, called the generalized reachable set, is a complete description of the problem's feasible region in objective function space. By providing two-dimensional slices of G_z, the method enables the decision maker to select a preferred solution. The development of computationally efficient convolution techniques for constructing G_z has made the GRS method a viable alternative to more traditional simplex based algorithms.

4.3 R-Optimality Concept (Nogin 1976)

A point $\bar{x}_o$ is said to be r-optimal if it is non-inferior with respect to all possible combinations of r of the objectives $\{f_1(\bar{x}_o), \ldots, f_m(\bar{x}_o)\}$, where $r \in \{1, \ldots, m\}$. Using the notion of seeking an r-optimal solution with the lowest value for r, a series of theorems can be proved which provide the groundwork for developing a solution procedure based on this concept. In essence, the procedure represents an intuitively appealing approach for reducing the number of candidate solutions.

4.4 Piecewise Linear Approximation Method (Polishchuk 1979)

This method both constructs a piecewise linear approximation, $S_{[a,b]}(\xi; t)$, of the non-inferior set in bicriteria convex programming problems and provides a metric δ from which to gauge the maximum possible error inherent in the approximation at each iteration (Figure 2).

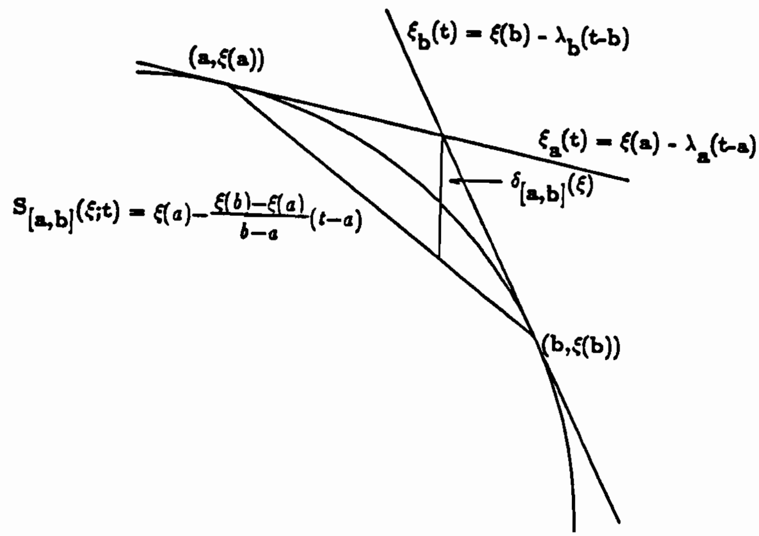


Figure 2: Piecewise linear approximation $S_{[a,b]}(\xi; t)$, maximum error metric $\delta_{[a,b]}(\xi)$, and the triangle containing non-inferior set ξ in interval $[a, b]$.

There are striking parallels between this method and the Non-Inferior Set Estimation (NISE) Method, developed independently by American researcher J. L. Cohon (1978), although the latter was designed strictly for linear problems, whereas the former is intended for the broader class of convex programming problems.

References

Bedel'baev, A. A., Yu. A. Dubov, and B. L. Shmul'yan. (1976). "Adaptive Decision Procedure in Multicriterion Problems. *Automation and Remote Control* no. 1 (January): 76-85.

Benayoun, R., J. de Montgolfier, J. Tergny, and O. I. Larichev. (1971). Linear Programming with Multiple Objective Functions: Step Method (STEM). *Mathematical Programming* no. 1: 366-375.

Benayoun, R., O. I. Larichev, J. de Montgolfier, and J. Tergny. (1971). Linear Programming with Multiple Objective Functions. The Method of Constraints. *Automation and Remote Control* 32, no. 8 (August): 1257-1264.

Bushenkov, V. A. and A. V. Lotov. (1980). Methods and Algorithms for Analyzing Linear Systems, by Constructing Generalized Sets of Attainability. *USSR Journal of Computational Mathematics and Mathematical Physics* 20, no. 5: 38-49.

———. (1982). *Metody postroeniya i ispol'zovaniya obobshchennykh mnozhestv dostizhimosti.* (Methods of Constructing and Using Generalized Reachable Sets.) Moscow: Computer Center of the USSR Academy of Sciences.

Cohon, J. L. (1978). *Multiobjective Programming and Planning.* New York: Academic Press.

Dubov, Yu. A. (1977). Resource Allocation Under Multiple Criteria. *Automation and Remote Control* 38, no. 12 (December): 1807-1814.

———. (1978). Multiple-Criterion Sequential Decision-Making Procedure. *Automation and Remote Control* 39, no. 10 (October): 1493-1497.

———. (1979). *Usloviya optimal'nosti v dinamicheskikh mnogokriterial'nykh zadachakh. Preprint.* (Optimality conditions in dynamic multicriterion problems. Preprint.) Moscow: Institute for Systems Studies.

Dubov, Yu. A. and B. L. Shmul'yan. (1973). Improvement of the Solution of Discrete Multicriterion Problem. *Automation and Remote Control* no. 12: 1953-1959.

Eiduk, Ya. Yu. (1981). Vektorno-relaksatsionnye algoritmy poiska kompromissnogo resheniya. (Vector-relaxation algorithm for seeking a compromise decision.) In *Metody i sistemy prinyatiya reshenii. Metody i modeli analiza reshenii,* 109-116. Riga: Riga Politechnic Institute.

————. (1983). Control of Complicated Systems in Case of Vectorial Indicator of Quality. In *Fourth Formator Symposium on Mathematical Methods for Analysis of Large-Scale Systems.* Proceedings of a Symposium held in Liblice near Prague, May 18-21, 1982, ed. Jiri Benes and Lubomir Bakule, 165-180. Prague: Academia.

Evans, G. W. (1984). An Overview of Techniques for Solving Multiobjective Mathematical Programs. *Management Science* 30, no. 11 (November): 1268-1282.

Hwang, C.-L. and A. S. M. Masud. (1979). *Multiple Objective Decision Making — Methods and Applications.* Lecture Notes in Economics and Mathematical Systems, *vol. 164.* Berlin: Springer-Verlag.

Kaplinskii, A. I. and A. S. Krasnenker. (1975). Multicriteria approach to the construction of multilevel stochastic optimization algorithms. (Collection title unknown.) 9, no. 4: 14-21.

————. (1977). O formirovanii dialogovykh algoritmov vektornoi optimizatsii. (On the formation of dialogue vector optimization algorithms.) *Avtomatika i vychislitel'naya tekhnika* no. 5: 32-37.

————. (1979). O strukture dialoga v protsessakh vektornoi optimizatsii dlya zadach sistem avtomaticheskogo proektirovaniya. (On a dialog structure in vector optimization processes for computer aided design problems.) *Avtomatika i vychislitel'naya tekhnika* no. 3: 37-43.

Khomenyuk, V. V. (1977a). *Optimal'nye sistemy upravleniya.* (Optimal control systems.) Moscow: Nauka.

————. (1977b). Primenenie kriteriya maksimal'noi effektivnosti v zadachakh vektornoi optimizatsii. (Application of the maximal effectiveness criterion in vector optimization problems.) In *Prikladnye metody teorii optimizatsii. Problemy vektornoi optimizatsii,* no. 13, Vladivostok: USSR Academy of Sciences Far East Scientific Center.

————. (1983). *Elementy teorii mnogotselevoi optimizatsii.* (Elements of the theory of multi-objective optimization.) Moscow: Nauka.

Korotkova, T. I. (1978). *O modifitsirovannykh funktsiyakh v mnogokriterial'nykh zadachakh.* (On modified functions in multi-criterial problems.) Moscow: Moscow Aviation Institute.

———. (1982). O vozmozhnosti resheniya zadach mnogokriterial'noi optimizatsii bol'shoi pazmernosti. (On the possibility of solving large dimension multi-criteria optimization problems.) In *Problemy matematicheskogo obespecheniya sistem avtomatizipovannogo proektirovaniya,* 13-16. Moscow.

———. (1983). Algorithm for the Solution of a Many Dimensional Multiple-Criterion Optimization Problem. *Automation and Remote Control* no. 3 (March): 350-358.

Krasnenker, A. S. (1975). The Method of Local Improvements in the Vector Optimization Problem. *Automation and Remote Control* no. 3 (March): 419-422.

Krasnoshchekov, P. C., V. V. Morozov, and V. V. Fedorov. (1979a). Decomposition in Planning Problems. I. *Engineering Cybernetics* 17, no. 3 (March-April): 1-10.

———. (1979b). Planning of Technological Systems of Multipurpose Assignment. *Engineering Cybernetics* 17, no. 4 (July-August): 5-11.

———. (1979c). Sequential Clustering in Internal Design Problems of Engineering Systems. *Engineering Cybernetics* 17, no. 5 (September-October): 1-7.

Lotov, A. V. (1972). Numerical Method of Constructing Attainability Sets for a Linear Control System. *USSR Journal of Computational Mathematics and Mathematical Physics* 12, no. 3: 279-283.

———. (1973). A Numerical Method of Studying the Optimal-Time Continuity in Linear Systems and the Solution of the Cauchy Problem for Bellman's Equation. *USSR Journal of Computational Mathematics and Mathematical Physics* 13, no. 5: 256-271.

Merkur'ev, V. V. and M. A. Moldavskii. (1979). A Family of Convolutions of a Vector-Valued Criterion for finding points in a Pareto Set. *Automation and Remote Control* no. 1: 87-97.

Mikhalevich, V. S. and V. L. Volkovich. (1982). *Vychislitel'nyi metod issledovaniya i proektirovaniya slozhnykh sistem.* (Computational method for research and design of complex systems.) Moscow: Nauka.

Moldavskii, M. A. (1980). Solution of Continuous Vector Optimization Problems. *Soviet Automatic Control* no. 2: 26-39.

Nogin, V. D. (1976). A New Way of Narrowing the Region of Compromises. *Engineering Cybernetics* no. 5: 7-12.

Polishchuk, L. I. (1978). Ob odnom semeistve dialogovykh protsedur prinyatiya resheniі. (On one family of dialogue decision making procedures.) In *Sistemy i metody obrabotki dannykh,* 76-86. Novosibirsk: USSR Academy of Science, Siberian Branch, Computer Center.

————. (1979). Kysochno-lineinaya approksimatsiya paretovoi granitsy vypuklykh dvukhkriterial'nykh zadach. (Piecewise linear approximation of the Pareto boundary of convex two criteria problems.) In *Modeli i metody issledovaniya ekonomicheskikh sistem,* 108-116. Novosibirsk: USSR Academy of Sciences, Siberian Branch.

————. (1980). Karty paretovoi granitsy porozhdennye rassloeniyami kriteriev. (Pareto boundary maps generated by partitioning criteria.) In *Metody analiza vzaimodeistviya v ekonomicheskikh sistemakh,* 78-100. Novosibirsk: Nauka.

————. (1981). Generalized Gradient Method in Interactive Optimization Procedures. *Automation and Remote Control* no. 5 (May): 644-651.

Polishchuk, L. I. and B. G. Mirkin. (1980). Mnogokriterial'nye zadachi ekonomikomatematicheskogo modelirovaniya i metody ikh resheniya. (Multi-criterial problems in economic-mathematical modeling and methods for their solution.) In *Modeli analiza dannykh i prinyatiya reshenii,* 4-58. Novosibirsk.

Popkov, Yu. S., B. L. Shmul'yan, N. V. Ikoeva, and A. B. Kabakov. (1974). Mnogokriterial'nyi podkhod dlya optimizatsii ocherednosti stroitel'stva v slozhivsheisya chasti goroda. (Multi-criterial approach to optimization of construction priorities in the existing part of a city.) In *Ispol'zovanie prikladnogo sistemnogo analiza v proektirvovanii i upravlenii razvitiem gorodov,* 166-179. Moscow: Stroiizdat.

Rastrigin, L. A. and Ya. Yu. Eiduk. (1985). Adaptive Methods of Multicriterial Optimization. *Automation and Remote Control* 46, no. 1 (January): 1-21.

Salukvadze, M. E. (1971a). Optimization of Vector Functionals I. The Programming of Optimal Trajectories. *Automation and Remote Control* no. 8 (August): 1169-1178.

————. (1971b). Optimization of Vector Functionals II. The Analytic Construction of Optimal Controls. *Automation and Remote Control* no. 9 (September): 1347-1357.

————. (1979). *Vector-valued Optimization Problems in Control Theory.* New York: Academic Press.

Sobol', I. M. and R. B. Statnikov. (1981). *Vybor optimal'nykh parametrov v zadachakh co mnogimi kriteriyami.* (Choosing optimal parameters in problems with many criteria.) Moscow: Nauka.

————. (1982). *Nailuchshie resheniya – gde ikh iskat'. Seriya matematika, kibernetika. 1.* (The best decisions – where to seek them. Mathematics, Cybernetics Series. 1.) Moscow: Znanie.

Steuer, R. (1986). *Multiple Criteria Optimization: Theory, Computation, and Application.* New York: John Wiley and Sons.

Velichenko, V. V. (1975). Sufficient Conditions Conditions for Absolute Minimum of the Maximal Functional in the Multi-Criterial Problem of Optimal Control. In *Optimization Techniques. IFIP Technical Conference.* Lecture Notes in Computer Science, ed. G. I. Marchuk, 220-225. Berlin: Springer-Verlag.

When English translations of Russian language publications exist, they are given in place of the original Russian citations. Thus, if a Russian title is shown, it can be assumed that an English translation is not available.

Multi-Objective Mathematical Problems. Algorithms and Software

V. C. Mikhalevich, V. L. Volkovich, L. F. Dargejko,
G. A. Dolenko, Yu. P. Tchaplinskij
V.M. Glushkov Institute of Cybernetics
Ukrainian Academy of Sciences
Kiev, USSR

The presence of contradictory goals, the necessity of consideration for social and economic consequences, a hierarchical nature of organization of the national economy and many other things cause a multi-objective character of economic-mathematical models of decision-making in design and management planning processes.

Many planning-management and development problems can be formulated as multi-objective mathematical programming problems in the form of the following models:

— discrete separable

$$\min(\max)\Big\{F_k(x) = \sum_{j=1}^{n} f_{kj}(x_j),\ k = \overline{1,M}\Big\} \tag{1}$$

where

$$x \in X = \prod_j X_j, \qquad X_j = \{x_{jl_j}\}, \quad l_j = \overline{1,M_j}$$

$$g_l(x) = \sum_j g_{lj}(x_j)\ R_l\ g_l^*, \quad R_l = \{\leq, \geq\}, \quad l = \overline{1,N}$$

— discrete separable without constraints

$$\min(\max)\Big\{F_k(x) = \sum_j f_{kj}(x_j), \quad k = \overline{1,M}\Big\} \tag{2}$$

$$x \in X = X_1 \times X_2 \times \ldots \times X_n,$$
$$X_j = \{x_{jl_j}\}, \qquad l_j = \overline{1,M_j},$$

— linear continuous

$$\min(\max)\Big\{F_k(x) = \sum_j c_{kj}x_j, \quad k = \overline{1,M}\Big\} \tag{3}$$

on a set of constraints

$$X = \{x:\ Ax\ R\ b\},$$

where

A is an (n, N)-dimensional matrix

b is an N-dimensional vector, $x = (x_1, x_2, \ldots, x_n)$

— linear integer

$$\min(\max)\left\{F_k(x) = \sum_j c_{kj}x_j, \quad k = \overline{1, M}\right\} \tag{4}$$

under constraints

$$a_l(x) = \sum_j a_{lj}x_j \ R_j \ b_l, \quad l = \overline{1, N}$$

$x_j > 0 \quad j = \overline{1, n}$ are integers,

$R_l \qquad (l = \overline{1, n})$ are any of relations $\{\leq, \geq\}$;

— linear with Boolean variables

$$\min(\max)\left\{F_k(x) = \sum_j c_{kj}x_j, \quad k = \overline{1, m}\right\} \tag{5}$$

under constraints

$$a_l(x) = \sum_j a_{lj}x_j \ R_l \ b_l, \qquad l = \overline{1, N}$$
$$x \in X = X_1 \times X_2 \times \ldots \times X_n, \qquad X_j = \{0, 1\},$$

$R_l \qquad (l = \overline{1, N})$ are any of relations $\{\leq, \geq\}$.

For solving the described problems a method called the method of constraints is solved. It defines, as a compromise, an efficient solution which corresponds to the DM's preference on a set of criteria from solution of the minimax problem (Mikhalevich and Volkovich, 1982)

$$\min_{x \in X} \max_{i \in I} \rho_i w_i(x) \tag{6}$$

where $I = \{1, \ldots, M\}$ is a set of criteria; $p_i \in R = \{\text{int } R_M : \sum_i p_i = 1\}$ are weight coefficients which define the DM's preference on a set of criteria within the quantitative scale; $w_i(x) = w(f_i(x))$ is a monotone normalizing transformation of initial criteria, $w \in [0, 1]$.

The transformation w measures a degree of deviation of value of the i-th criterion from the optimal admissible value. $w_i(f_i(x))$ will be called the function of relative losses of the i-th criterion on an admissible domain.

The solution of the problem (6) is reduced to the solution of a more simple problem

$$\min k_0, \tag{7}$$
$$p_i w_i(x) \leq k_0 \qquad x \in X \tag{8}$$

Inequalities (8) mean that relative losses with respect to all criteria must not exceed a certain value k_0; it is this that gave rise to the name of the method.

Weight coefficients p_i depend upon the desired level of values of criterion functions and this dependence may be represent in form

$$p_i(f^r) = \frac{\prod_{j \neq i} w_j^r}{\sum_q \prod_{j \neq q} w_j^r} \tag{9}$$

and is interpreted in the following manner: the nearer the desired level to the best value, i.e. the smaller the value w_i, the more important is this criterion. Thus the weight coefficients represent the non-equivalence of various criteria for the DM, i.e. his preference.

This method does not depend on the form of criterion functions and a set of admissible solutions. It is only required that for each type of problems there should be efficient ways of testing the compability of the set of inequalities (8) on a given admissible domain D_0.

For problems (1)–(5) the algorithms of testing the compability of the problem (7)–(8) are developed based on the methodology of sequential analysis and elimination of variants (Volkovich et al., 1984).

For discrete multi-objective mathematical programming problems (1), (2), (4), (5) the algorithms of testing the compatibility of constraints (8) for each value of parameter k_0 are developed.

For the problem (1) a set of inequalities has the form

$$f_i(x) \leq (\geq) f_i^*(k_0), \tag{10}$$

where $f_i^*(k_0)$, $i = 1, \ldots, M$ are defined for the given value k_0 $[k_{min}, k_{max}]$ by relations

$$f_i^*(k_0) = f_i^0 + (-)\frac{k_0}{p_i}(f_{imax} - f_{imin}).$$

Denote $x^i = \{\arg\min f_i(x) = \sum_j f_i(arg \min f_{ij}(x_j))\}$, $x/x_j = (x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$ — is a vector of solution without the j-th component. Then we eliminate the elements x_j for which the inequalities

$$f_{ij}(x_j) \leq f_i^*(k_0) - f_i(x^i/x_j) = f_i^0(k_0) - \sum_j f_{ij}(x_j^i) \tag{11}$$

are not satisfied. The eliminated elements cannot form the solution satisfying inequalities (10).

For the problem (2), apart from (11), there should be satisfied at each step the inequalities (Volkovich and Dargejko, 1972)

$$g_{lj}(x_{jl_j}) \leq g_l^* - g_l(x^l/x_j), \qquad \forall\, l = \overline{1,N}, \quad j = \overline{1,n}. \tag{12}$$

The non-fulfillment of inequalities (11) and (12) is a sufficient condition for elimination of the elements, and in this case the elements which do not enter into any solution admissible by criteria and constraints are eliminated.

For the problem (4) a set of equations (8) has the form

$$\begin{aligned} &\frac{p_i}{f_{imax} - f_{imin}}\Big(\sum_j c_{ij}x_j - f_{imin}\Big) \leq k_0, \quad i = \overline{1,M}, \\ &x_j \in [d_{j(l)}, d_{j(u)}], \qquad\qquad\qquad \forall j = \overline{1,n}, \end{aligned} \tag{13}$$

x_j — are integers satisfying inequalities.

Let $R_l = \{\leq\}$. Then the algorithm of the method of constraints consists in eliminating solutions which do not satisfy the system (13) by narrowing the bounds of the change of variables $x_j[d_{j(l)}, d_{j(u)}]$ in the following manner

$$\begin{aligned} d_{j(u)}^{(k+1)} &= \min\Big\{d_{j(u)}^{(k)}, [d_{j(u)}^{(k+1)*}]\Big\}, \\ d_{j(l)}^{(k+1)} &= \max\Big\{d_{j(l)}^{(k)},]d_{j(l)}^{(k+1)*}[\Big\}, \end{aligned}$$

where $[\cdot]$ is the integer part of the number, $]\cdot[$ is the least integer not less than the given one.

$$\begin{aligned} d_{j(u)}^{(k+1)*} &= \min_{a_{ij}>0}\left\{\frac{1}{a_{ij}}\Big(\tilde{b}_i - \sum_{\substack{l\in j_i^+ \\ l\neq i}} \tilde{a}_{il} d_{l(l)}^{(k)} - \sum_{\substack{l\in j_i^- \\ l\neq i}} \tilde{a}_{il} d_{l(u)}^{(k)}\Big)\right\} \\ d_{j(l)}^{(k+1)*} &= \max_{a_{ij}<0}\left\{\frac{1}{a_{ij}}\Big(\tilde{b}_i - \sum_{\substack{l\in j_i^+ \\ l\neq i}} \tilde{a}_{il} d_{l(l)}^{(k)} - \sum_{\substack{l\in j_i^- \\ l\neq i}} \tilde{a}_{il} d_{l(u)}^{(k)}\Big)\right\} \\ j_i^+ &= \{j : \tilde{a}_{ij} > 0,\ j = \overline{1,n}\} \\ j_i^- &= \{j : \tilde{a}_{ij} < 0,\ j = \overline{1,n}\} \\ \tilde{a}_{ij} &= \begin{cases} a_{ij}, & i \in L \\ c_{ij}, & i \in I \end{cases} \\ \tilde{b}_i &= \begin{cases} b_i, & i \in L \\ f_{imin} + \dfrac{k_0}{p_i}(f_{imax} - f_{imin}), & i \in I \end{cases} \end{aligned}$$

$(I \cup L)$ — is a set of indices of criteria and constraints left at the k-th step after elimination of incidental constraints for which the inequality

$$\sum_{l\in j_i^+} \tilde{a}_{il} d_{l(u)}^{(k)} + \sum_{l\in j_i^-} \tilde{a}_{il} d_{l(l)}^{(k)} < \tilde{b}_i$$

is fulfilled.

For the problem (5) a system of equations is similar to relations of the problem (4) and the algorithm of its solution is based on eliminating the values of components of the vector of solutions x_l for which the inequality

$$c_{il}\tilde{x}_l > f_{il} - \sum c_{ij}\tilde{x}_{j(l)}$$

is fulfilled if only for one $i \in I$, or

$$a_{il}\tilde{x}_l > b_i - \sum_{j\neq l} a_{ij}\tilde{x}_{j(l)}$$

if only for one $i \in L$. Here $\tilde{x}_{j(l)}$ is the best of the values of components of the vector of solutions by the i-th criterion and by the i-th constraint respectively, left in the reduced set.

The process of the test of described inequalities continues until their right-hand sides stop to change, and the process of narrowing the bounds of variables continues until the number of elements left in the set is available for direct sort out.

A system of constraints (8) in case of the problem (3) will be described by relations

$$\begin{aligned} &\sum_j c_{ij}x_j \geq \sum_j c_{ij}x_j - \frac{k_0}{p_i}\Big(\sum_j c_{ij}x_j^{i(u)} - \sum_j c_{ij}x_j^{i(l)}\Big) \\ &\sum_j a_{l_j}x_j \leq b_l, \qquad l \in L, \qquad d_{j(l)} \leq x_j \leq d_{j(u)}, \qquad j = \overline{1,n}. \end{aligned} \tag{14}$$

By using the method of constraints we must find a unique solution x^*, for which the system of relations (14) is compatible under the minimal value of parameter k_0. This may be done by solving the following linear programming problem

$$\min \quad x_{n+1} = k_0$$

under constraints

$$\begin{aligned} &\sum_j d_{l_j}x_j + d_{l_{n+1}}x_{n+1} \leq d_l \\ &\sum a_{l_j}x_j \leq b_l, && l \in L, \\ &d_{j(l)} \leq x_j \leq d_{j(u)}, && j = \overline{1,n}, \\ &0 \leq x_{n+1} \leq \frac{1}{M}, \end{aligned}$$

where

$$d_{ij} = \begin{cases} -p_i c_{ij}, & i \in I_1, \\ p_i c_{ij}, & i \in I_2, \quad I_1 \cup I_2 = I \end{cases}$$

$$d_{i(n+1)} = \begin{cases} -\sum_j c_{ij}x_j - \sum_j c_{ij}x_j^{i(l)}, & i \in I_1 \\ \sum_j c_{ij}x_{j(u)}^{i} - \sum_j c_{ij}x_j, & i \in I_2 \end{cases}$$

$$d_l = \begin{cases} -\sum_j c_{ij}x_j, & i \in I_1 \\ \sum_j c_{ij}x_j, & i \in i_2 \end{cases}$$

The described problem can be solved by standard simplex-method.

In case of the incompatible system of constraints there arises a need for solving the problem of model parameter control for attaining for compatibility of the represented system of constraints; such problem, according to (Dolenko and Tchaplinskij, 1988), is interpreted as the system optimization problem. For solving the system optimization problem some algorithm were developed. Their main goal is to choose the most desirable direction of the search for solution of the problem of testing the fulfillment of the given requirements on the basis of the model being formed and,in case of their nonfulfillment, to construct a new model according to the additional bounded domain of model parameter variations being constructed and conditions of coordination of requirements D^G and the domain of admissible solutions with respect to criterion functions in which the specified requirements are met. First of all we single out the requirements (constraints) which describe the domain (directive domain) of the desired (the most acceptable for DM) solutions of the problem (6). In the linear case such requirements may be represented in the form of the following domains

$$\begin{aligned}
D_0^G &= \{x : \; x_j = x_j^{*(G)}\}, \\
P_0^G &= \{x : \; x^*_{j(l)} \le x_j \le x^*_{j(u)}, \quad j = \overline{1,n}\}, \\
D_r^G &= \{x : \; \sum_j a^0_{ij} x_j \le u^0_i, \quad i \in Q^G\} \\
D_f^G &= \{f : \; f_i \ge f_i^*, \quad i \in I_1, \quad f_i \le f_i^*, \quad i \in I_2\} \\
P_f^G &= \{f : \; f^*_{i(l)} \le f_i \le f^*_{i(u)}, \quad i \in I\} \\
D_A^G &= \{x : \; x - A^0 x = u^0\}
\end{aligned}$$

A^0 —is a productive matrix of direct costs.

The domain od admissible solutions is described by the set D^0

$$D_0 = \left\{x : \; \sum_j b^0_{ij} x_j \le b^0_i, \quad i \in Q\right\}$$

We assume that there may be the variation of parameters according to possible cases of contradictory interaction of the described systems of constraints of the domain D_0, D_A^G, P_f^G and D_f^G.

The test of compatibility of requirements of the domain in the domain D_0 is performed either by direct substitution of $x = x^{*(G)}$ into the system of constraints of the domain D_0 for D_0^G and or on the basis of the procedure of elimination of knowingly inadmissible solutions or some linear programming method for domains P^G and D_r^G or by testing the compatibility of these domains for the point which is the solution of the problem (6) for $x \in P^f$ where P^f approximates the domain od admissible solutions of the system D_f^G, P_f^G.

In case of empty intersection of D^G and D_0 various casesare possible of mutual disposition of the domain D^G and the domain D_0 with respect to criterion functions which in the general case may be the following ones:

1. All points of the directive domain D^G have the best values with respect to all criteria F_i, $i \in I$ as compared to values attained at points of the domain corresponding to them in preference, i.e., the complete agreement.

2. For any point of the directive domain D^G and the domain of admissible solutions D_0 there exists the point with the best values with respect to all criteria simultaneously, i.e., directive requirements are not in agreement with the goals of the considered system specified by a set of criterion functions.

3. Only a part of points of the directive domain gives the improvement of values with respect to all criteria simultaneously, i.e., the requirements from D^G are only in a partial agreement with the goals of the given system.

With regard for the realized variant of disposition and with regard for a form of the directive domain we may single out the constraints of the domain D_0 which obstruct the compatibility and agreement of solutions from D^G and the domain of admissible solutions; they will be called essential constraints.

In case of directive domains D_0^G, D_f^G, P_f^G relations which are violated with substitution x^* will be essential constraints. A set Q^0 of indices of essential constraints for P^G and D_r^G there may be considered the cases with account for the specified set of criteria and without consideration for objective functions.

With consideration for criteria after definition of the realized variant of agreement we single out a set of points called the domain of capture,which approximates the domain when the first variant is realized,or the domain of capture X^G, $X^G \subseteq D^G$ (containing points which have the best values with respect to all criteria simultaneously as compared to solutions of the domain D_0) — when the third variant is realized.

If system optimization problem is solved without consideration for the set of criterion functions then singling out the essential constraints depend upon the domain of capture X^G, choosen by the DM, which may be described by the parallelepiped P^{G*} or by the set D, $D \subseteq D^G$ or by the point x^* (Dolenko and Tchaplinskij, 1988).

For the specified domain of capture to belong to the variable model a system of constraints describing the domain is constructed.

Thus, if the domain D^D of capture is specified in terms of the domain D_0^G, i.e., by the desired point $x^{*(G)}$, then these constraints will be written in the form

$$\sum_j x_j^{*(G)} \Delta b_{pj} - \Delta b_p \leq b_p^0 - \sum_j b_{pj}^0 x_j^{*(G)}, \quad p \in Q^0 \tag{15}$$

$$\begin{aligned} &\Delta b_{pj} > -b_{pj}^0 \quad \text{if} \quad b_{pj}^0 > 0, \quad \Delta b_{pj}^0 < |b_{pj}^0| \quad \text{if} \quad b_{pj}^0 < 0, \\ &\Delta b_{pj} > -b_p^0 \quad \text{if} \quad b_p^0 > 0, \quad \Delta b_{pj}^0 < |b_p^0| \quad \text{if} \quad b_p^0 < 0, \end{aligned} \tag{16}$$

The relations (15) define the domain of variations P^{G_0} which allows the point $x^{*(G)}$ to be made admissible in the new model, and relations (16) describe a physical domain P_f which is necessary for a physical sense of constraints on sets Q^0 not to be violated. Then the domain P of parameter variations defined by interaction of the domain P^{G_0} and P_f will be written in the form of a system of constraints (15), (16).

Starting from technical-economic possibilities of the formed problem we construct the domain P_0 of admissible parameter variations of constraints of the set Q^0. Here the constraints of the domain P_0 on parameter variations will be described by relations constructed before the stated problem was solved or specified by the DM in the process of execution of the given system optimization procedure and defined either by two-sided or connecting constraints on parameter variations or by their combinations.

To find the possibility of the change of problem model for meeting the requirements from D^G and, therefore, the possibility of solution of the system optimizition problem itself as well as the problem (6) we construct the intersection of the domain P of parameter variations of constraints of the set Q^0 and the domain P_0 of admissible variations of these parameters. If $P \cap P_0 \neq \emptyset$ then the domain of variations of model parameters will be restricted and this makes it possible to solve the problem of construction of a new model in which the requirements from the domain D^G are met. If $P \cap P_0 = \emptyset$ then in this case it is necessary either to change the constraints P_0 or the DM should specify again his requirements.

As the DM changes his requirements for $P \cap P_0 \neq \emptyset$ it is sufficient that there exist a non-empty intersection of the domain D_x described by relations of the form

$$\sum_j (b_{ij}^0 + \Delta b_{ij}'')x_j \leq b_i^0 + \Delta b_i'', \quad i \in Q^0, \quad \Delta b_{ij}'', \Delta b_i'' \in P_0$$

and the directive domain D^G with consideration for the variant of disposition with respect to D^G or without it; in order that some point or domain exist, with respect to which the solution of the system optimization problem can be continued, it is necessary that there exist a non-empty intersection of the domain D_x described by relations cutting off inadmissible solutions from the specified domain P_0 and the domain D^G.

For correction of the domain P_0 we use the information about the constructed domain P.

The problem of the choice of variations of parameters under non-empty intersection of domains P and P_0 is reduced to the system optimization problem in which the costs related to variations of parameters of the model $C(\Delta)$

$$\min \quad C(\Delta)$$

under constraints

$$\Delta \in P \cap P_0$$

serve as criteria.

If it is impossible to construct a cost function, the choice problem is formed as a multi-objective problem where each parameter is an individual criterion which according to a physical essence may be maximized or minimized

$$f^p = \{\Delta_{pj}, \quad p \in Q^0, \quad j = \overline{1,n}, \quad \Delta p, p \in Q^0\}$$

By conditions of construction the new model of the formulated problem provides the realizability of requirements from D^G and DM and the existence of solution with criterion values not worse than the desired ones.

On the basis of the described algorithms a software for solution of multi-objective problems is constructed. It is a dialog system which, on the one hand, enables the DM in the process of working out solutions to use a considerable arsenal of non-formalized factors according to his experience and intuition and, on the other hand, the system algorithms provide the possibility to process a great amount of the analysis a whole gamma of solutions of the formulated problem.

Such software comprises the multi-objective optimization dialog system DISMOP and the system optimization dialog complex DIKSOP.

The DISMOP is designed for supporting interactive operation with the model developed by experts or analysts before hand and computerized using a mathematical tool of solving multi-objective optimization problems based on the method of constraints and man-machine procedure constructed on the concept of optimization by a specified point.

The purpose of the DISMOP system is to compute and display some chosen efficient solutions. This choice of efficient solutions is easily controlled by a user and provides the possibility of obtaining any efficient solution in a consistent admissible domain which is suitable for user (DM). The DISMOP can be used by analysts which desire to analyze the model being formed or by the DM for choosing the best solution of the stated multi-objective problem.

The DISMOP makes it possible to analyze and solve real problems whose mathematical models may be represented in the form (1)–(5).

This dialog system supports various forms of representation and editing of the mathematical model of the problem being solved. The interaction between the user and computer enables the user:

- to scan the best and the worst values of criterion functions with respect to each criterion individually;
- to obtain and display for the analysis the efficient solution corresponding to various forms of specifying the preference;
- to specify preferences in the form:
  - explicitly in terms of weight coefficients,
  - of the desired values of criteria,
  - of percentage deviations from the best values of criteria;
- to obtain graphic representation of accumulated efficient points.

For the most clarity and convergence of the constructed algorithms at each step of problem solving the possible and narrowed bounds of criterion values are displayed to the user (DM).

The dialog complex DIKSOP is the software realization of algorithmic support of solution of the system optimization problem. The DIKSOP makes it possible to solve the following problems: the forming of the mathematical models and solution of system optimization problems for various classes of models; organization of the DM-computer interaction for the real-time input of heuristic information in the process of problem solution; organization of a service subsystem of the decision-making process.

The DIKSOP is constructed in such a way that it can operate with the information being in the main memory as well as with the information fed from external computers, and it can interact with earlier developed programs or application program packages.

The developed program system affords the interaction with programs for which input and output parameters are described in the format of the application program package of multi-objective optimization DISMOP, of commercial application program packages of MPSX type, of the library of mathematical programming IMSL.

The DIKSOP can be used in the following applications: support of the HELP mode according to the types of problems being solved, according to the system optimization problem being solved and according to the individual procedure of system optimization; organization of the mode of scanning of a chain of system optimization procedures in the process of decision making; storage of specified domains of capture before their correction and support of their interactive analysis from the viewpoint of objective function values on these domains and a degree of violation of essential constraints corresponding to these domains, a degree of incompability of the domains P and P_0; the possibility of stopping of the computational process for respecify in the domain of capture; organization of restart of the program complex.

The described algorithmic and program support of solving multi-objective problems has been as a software of the process of working out planning-management and development decisions in large organization-economic systems (the choice of a production program, the choice of a structure of complex engineering systems based on the given components).

References

Dargejko, L. F. (1976). Method of constraints in vector optimization problems. *Kibernetika i vychislitel'naja technika.* N 31, pp. 45–53.

Dolenko, G. A., Tchaplinskij, Yu. P. (1988). Man-machine system optimization procedure in planning. Decision making in integrated control systems. Kiev: V. M. Glushkov Institute of Cybernetics of the Academy of Sciences of the Ukr. SSR. pp. 5–11.

Mikhalevich, V. S. (1962). Algorithms of sequential analysis of variants and their application. *Kibernetica.* N 1, pp. 12–18.

Mikhalevich, V. S., Volkovich, V. L. (1982). Computational methods in study and design of complex systems. Kiev : Naukova dumka, 285 p.

Volkovich, V. L., Dargejko, L. F. (1972). On one algorithm of the choice of trade-off decision for linear criteria. *Kibernetika*, N 5, pp. 133–136.

Volkovich, V. L., Tchaplinskij, Yu. P. (1987). System optimization algorithms for specifying a directive domain by balance relations. *Avtomatika.* N 6, pp. 45–53.

Volkovich, V. L., Voloshin, A. F., Gorlova, T. M., Dargejko, L. F., Maltsev, V. V. (1984). Methods and algorithms of computer-aided design of complex control systems. Kiev : Naukova Dumka, 214 p.

Satisficing Trade-off Method for Problems with Multiple Linear Fractional Objectives and its Applications

Hirotaka Nakayama
Department of Applied Mathematics
Konan University
8-9-1 Okamoto, Higashinada, Kobe 658, Japan

Abstract

The aspiration level approach to multi-objective decision problems is observed to be very effective in many real problems. The satisficing trade-off method is one of them, and has been being applied to several kinds of real problems. Among them, the problem, in which some ratio are to be maximized or minimized, is of fractional programming. For example, in bond trading, we have several linear fractional objective functions to be maximized or minimized such as the average direct yield, the average yield to maturity, the average effective yield and so on. In the application of the satisficing trade-off method to problems with multiple fractional objective functions, we have to solve the so-called a generalized fractional programming problem. In this paper, it will be shown that the computer program using Dinkelbach type algorithm is developed, and effectively applied to bond trading problems and cement blending problems.

1 Introduction

In the past few decades, theoretical aspects of multi-objective decision problems has been remarkably investigated, and several kinds of solution techniques has been developed. It seems that it is the time to apply those methods to real problems. Among them, from a practical viewpoint, the aspiration level approach is very attractive according to the following reasons:

1. it requires the decision maker to answer only his/her aspiration level, which is very easy to answer;

2. it does not require any consistency of the decision maker's judgment;

3. it makes the trade-off (or equivalently balancing) among the objectives very easy.

The second point above is very important, because it can not be expected in many real situations that decision makers do not behave consistently in such a manner that many mathematical theory insist (for example, the transitivity of indifference does not hold in many cases (Tversky, 1969). Above all, most of all mathematical theories do not reflect the change of attitude of decision makers. In the decision process such as interactive programming, since new information becomes available to decision makers one after another, decision makers usually change their attitude during the decision process. The aspiration level approach allows decision makers to change their attitudes, which are represented by aspiration levels.

Therefore the aspiration level approach seems the most promising from a viewpoint of human factors. Now, the thing what we, systems-scientists, have to do is to improve effectively information process from the computer side. As an example, for problems formulated as multi-objective programming, we can cite the use of sensitivity (parametric) analysis in trading off (Korhonen-Wallenius, 1987, Nakayama, 1987). In addition, it is important to develop practical techniques for solving various real problems. The author is recently engaged in applications of the satisficing trade-off method to several kinds of real problems. In this paper, based on his experiences, some devices for problems with linear fractional objective functions are reported along with their applications to bond portfolio problems and material blending problems in cement production.

2 Satisficing trade-off method for problems with linear fractional objective functions

2.1 Satisficing trade-off method (Nakayama, 1984)

First, we shall review the procedure of the satisficing trade-off method briefly. The problem to be considered is as follows:

$$\begin{array}{lll} \text{(P)} & \text{Maximize} & F(x) = (F_i(x), \ldots, F_r(x)) \\ & \text{subject to} & x \in X \end{array}$$

where the constraint set X may be represented by some constraint functions $g_j(x) \leq b_j$ $(j = 1, \ldots, m)$. The algorithm of the satisficing trade-off method is summarized as follows:

Step 1. (setting the ideal point) The ideal point $F^* = (F_1^*, \ldots, F_r^*)$ is set, where F_i^* is large enough, for example, $F_i^* = \max \{F_i(x) \mid x \in X\}$. This value is fixed throughout the following process.

Step 2. (setting the aspiration level) The aspiration level $\bar{F}_i^k$ of each objective function F_i at the k-th iteration is asked to the decision maker. Here $\bar{F}_i^k$ should be set in such a way that $\bar{F}_i^k < F_i^*$. Set $k = 1$ for the first iteration.

Step 3. (weighting and finding a Pareto solution by the Min-Max method) Set

$$w_i^k = \frac{1}{F_i^* - \bar{F}_i^{k'}} \tag{2.1}$$

and solve the Min-Max problem

$$\min_{x \in X} \max_{1 \leq i \leq r} w_i^k |F_i^* - F_i(x)| \tag{2.2}$$

Let x^k be a solution of (2.2).

Step 4. (trade-off) Based on the value of $F(x^k)$, the decision maker classifies the criteria into three group, namely,

(i) the class of criteria which he wants to improve more,

(ii) the class of criteria which he may agree to relaxing,

(iii) the class of criteria which he accepts as they are.

The index set of each class is represented by I_I^k, I_R^k, I_A^k, respectively. If $I_I^k = \emptyset$, then stop the procedure. Otherwise, the decision maker is asked his new acceptable level of criteria $\bar{F}_i^k$ for the class of I_I^k and I_R^k. For $i \in I_A^k$, set $\bar{F}_i^k = F_i(x^k)$. Go back to Step 3.

Remark 2.1.

In LP type problems, we can use parametric analysis in trade-off very effectively. This device enables us to trace Pareto surface without solving any additional Min-Max problem (Nakayama, 1987, Korhonen-Wallenius, 1987). For bond portfolio selection and material blending problems in cement production, some of objective functions are of linear fractional form as will be seen later. Therefore the auxiliary min-max problem (2.2) becomes a linear fractional programming, we need some device for solving it.

2.2 An algorithm for linear fractional Min-Max problem

Let each objective function be of the form $F_i = p(x)_i/q_i(x)$ $(i = 1, \ldots, r)$ where p_i and q_i are linear with respect to x. Then since

$$F_i^* - F_i(x) = \frac{F_i^* q_i(x) - p_i(x)}{q_i(x)} := \frac{f_i(x)}{g_i(x)},$$

the auxiliary Min-Max problem (2.2) becomes a kind of linear fractional Min-Max problem. For this kind of problem, several methods have been developed. Here we shall use a Dinkelbach type algorithm (Borde-Crouzeix, 1987, Ferland-Potvin, 1985) as is stated in the following:

Step 1: Let $x^0 \in X$. Set $\theta^0 = \max_{1 \leq i \leq r} f_i(x^0)/g_i(x^0)$ and $k = 0$.

Step 2: Solve the problem

$$(\mathrm{P}_k) \qquad T_k(\theta^k) = \min_{x \in X} \max_{1 \leq i \leq r} (f_i(x) - \theta^k g_i(x))/g_i(x^k)$$

Let x^{k+1} be a solution to (P^k).

Step 3: If $T_k(\theta^k) = 0$ then stop: θ^k is the optimal value of the given Min-Max Problem, and x^{k+1} is the optimal solution.

Step 4: If $T_k(\theta^k) \neq 0$, take $\theta^{k+1} = \max_{1 \leq i \leq r} f_i(x^{k+1})/g_i(x^{k+1})$. Replace k by $k+1$ and go to Step 2.

Note that the problem (P_k) is the usual linear Min-Max problem. Therefore, we can obtain its solution by solving the following equivalent problem in a usual manner:

$$
\begin{array}{lll}
(\mathrm{Q}_k) & \text{Minimize} & z \\
 & \text{subject to} & (f_i(x) - \theta^k g_i(x))/g_i(x^k) \leq z, \quad i = 1, \ldots, r \\
 & & x \in X
\end{array}
$$

3 Applications

3.1 Material blending in cement production

Cement is produced by blending, crushing and burning several raw material stones such as lime, clay, silica, iron and so on. The provided raw material stones change from time to time. In order to keep the quality of cement at the desired level, the supply of each raw material stone is controlled depending on the change of chemical ingredient of raw material stones. Each factory usually produces several kinds of cement, and therefore the desired property of cement is various. It is needed to develop an effective method which leads to an appropriate solution for blending associated with the change of the kind of cement and that of ingredient of raw material stones. Usually, this material blending is implemented by a minicomputer linked with ingredient analyzers. To this aim, the satisficing trade-off method seem to work well. Some results of our experiments will be shown in the following.

	CaO	SiO_2	Al_2O_3	Fe_2O_3
lime	50.8	5.1	1.1	0.6
clay	4.0	63.6	15.7	7.2
silica	1.1	85.7	6.7	2.7
iron	1.5	27.4	3.3	67.8
another	12.9	41.5	15.1	7.7

Table 1. An example of ingredient of raw material stones

The table 1 represents one of examples for chemical ingredients of each raw material stones. The criteria to be considered usually are the hydraulic modulus, the silica modulus and the iron modulus. Let x_i denote the amount of the i-stone to be used, and let C_i, S_i, A_i and F_i respectively denote the amount of CaO, SiO_2, Al_2O_3 and Fe_2O_3 contained in the i-stone. Each criterion is given by

i) hydraulic modulus:

$$
\mathrm{HM} = \sum_{i=1}^{5} C_i x_i / \sum_{i=1}^{5} (S_i + A_i + F_i) x_i
$$

ii) silica modulus:

$$\mathrm{SM} = \sum_{i=1}^{5} S_i x_i / \sum_{i=1}^{5} (A_i + F_i) x_i$$

iii) iron modulus:

$$\mathrm{IM} = \sum_{i=1}^{5} A_i x_i / \sum_{i=1}^{5} F_i x_i$$

The following constraints are imposed:

1. total amount to be blended:

$$\sum_{i=1}^{5} x_i = x_0$$

2. limitation of material supplier:

$$L_i \leq x_i \leq U_i \qquad (i = 1, \ldots, 5)$$

For material blending in cement production, many objective functions are to be attained at the desired levels. Since the desired levels are flexible to some extent, the decision maker is asked to answer his desirable level with its allowable tolerance range for each criterion. After this, each criterion is treated as two objective functions, for example,

$$\mathrm{HM} \geq \overline{\mathrm{HM}} - \varepsilon, \qquad \mathrm{HM} \leq \overline{\mathrm{HM}} + \varepsilon$$

where $\overline{\mathrm{HM}}$ and ε denote the desirable level (target) and the allowable range for the hydraulic modulus, respectively.

As an example, we will show a result for the data in Table 1. At present, many factories operate the blending by use of the goal programming in which the above three criteria are considered. For the normal cement, the desired level of each criterion is (HM, SM, IM) = (2.08, 2.59, 1.83).

In the goal programming, the deviation of each criterion from its target (originally it is of linear fractional form) is transformed into some linear objective function so that the simplex code for LP may be applied. Due to this, the weighting for each objective function is very difficult, because people can not know the quantitative relation between the weight and the corresponding solution. Even if the decision maker wants to improve some of criteria and increase the corresponding weight, the obtained solution often causes other criteria worse too much. Many trial-and-errors are usually needed in order to get an appropriate weight.

By using the satisficing trade-off method, we can avoid this trouble. Since the obtained solution by the satisficing trade-off method has 'equality' to attain the target for each criterion, the setting or change of the target and allowable range for each criterion is easy.

<u>Case 1:</u>

For our problem, we set the allowable error to be 0. Then we obtain the solution which attains the exactly desired level of each criterion.

	Pareto sol.	Target	Allowable Range
HM	2.080	2.080	0.0
SM	2.590	2.590	0.0
IM	1.830	1.830	0.0

$x_1 = 573.713,\quad x_2 = 26.458,\quad x_3 = 23.136,\quad x_4 = 1.693,\quad x_5 = 95.000$

Case 2:

In many factories, the cost for blending is not taken into account up to now. Next, we shall show results for cases in which the cost is taken into account:

iv) cost: $P = \sum_{i=1}^{5} p_i x_i$

For the same aspiration level and the allowable error as the above, we set the aspiration level of cost to be 15.0 with the allowable range 3.0. The result is given by the following:

	Pareto sol.	Asp. Level (Target	Range)	Lowest	Highest
F1 (tar)	2.1839	2.0800	.0000	.8706	3.1961
F2 (tar)	2.6968	2.5900	.0000	1.3472	5.8837
F3 (tar)	1.8017	1.8300	.0000	.3894	2.1598
F4 (min)	15.6700	15.0000		7.2000	52.9500

$x_1 = 599.8356,\quad x_2 = 100.0000,\quad x_3 = 3.5264,\quad x_4 = 2.5906,\quad x_5 = 14.0474$

If the decision maker agree with the increase the aspiration level of cost to 25.0, we can get the following solution by the satisficing trade-off method

	Pareto sol.	Asp. Level (Target	Range)	Lowest	Highest
F1 (tar)	2.0800	2.0800	.0000	.8706	3.1961
F2 (tar)	2.5900	2.5900	.0000	1.3472	5.8837
F3 (tar)	1.8300	1.8300	.0000	.3894	2.1598
F4 (min)	22.3000	25.0000		7.2000	52.9500

$x_i = 588.8332,\quad x_2 = 100.0000,\quad x_3 = 0.4293,\quad x_4 = 2.3431,\quad x_5 = 28.3944$

This solution shows that is costs at least 22.3 to attain the requirement aspiration level. Recall that the cost for the solution of the case 1) is 56.662. It can be seen that taking the cost into account, we can get a solution with the cost less than half of case 1), while other criteria attain at the desirable levels.

Case 3:

If the criteria are flexible to some extent around the target, we can get another solution with less cost. The following is the result for the case with the allowable range 0.1 around the target for each criterion.

	Pareto sol.	Asp. Level (Target	Range)	Lowest	Highest
F1 (tar)	2.1794	2.0800	.1000	.8706	3.1961
F2 (tar)	2.6894	2.5900	.1000	1.3472	5.8837
F3 (tar)	1.7302	1.8300	.1000	.3894	2.1598
F4 (min)	14.9963	15.0000		7.2000	52.9500

$$x_1 = 600.3790, \quad x_2 = 100.0000, \quad x_3 = 4.5450, \quad x_4 = 3.4324, \quad x_5 = 11.6436$$

It is seen that we can get a solution with less that one third cost of the case 1), while other criteria remain within an allowable error. As can be seen through the above experiments, we can easily obtain a solution as we desire from a total viewpoint. Unlike the traditional goal programming, since we search the solution by adjusting the aspiration level, which is very easy to answer, and has a direct quantitative relation with the corresponding Pareto solution, the satisficing trade-off method seems very operationable. Some cement company is now going to build the method in its material blending system.

3.2 Bond portfolio problems

Bond traders are facing almost every day a problem which bonds and what amount they should sell and/or buy in order to attain their customers' desires. The economic environment is changing day by day, and sometimes gives us a drastic change. Bond traders have to take into account many factors, and make their decisions very rapidly and flexibly according to these changes. The number of bonds to be considered is usually more than 500, and that of criteria is about ten as will be shown later. The amount of trading is usually more than 1,000 million yen, and hence even a slight difference of portfolio combination might causes a big difference to profit or loss. This situation requires some effective method which helps bond traders following faithfully their value-judgment on a real time basis not only mathematically but also in such a way that their intuition fostered by their experiences can be taken in.

The traditional mathematical programming approach with a single objective function can not take in the value-judgment and intuition of bond traders so easily in a flexible manner for the changes of environment. We shall show that some interactive multi-objective programming method fits to this purpose.

Returns from bonds are the income from coupon and the capital gain due to price increase. Bond portfolio problems are to determine which bonds and what amount the investor should sell and/or buy taking into account many factors, say, expected returns and risk, the time needs money for another investment, and so on. The problem is formulated as an optimization problem with linear fractional multi-objective functions. See the details, for example, (Nakayama 1988).

4 Concluding remarks

The stated method is now trying to build in a trading system in a security company and in a blending system in cement production of an industrial company in Japan. Of course, the role of the method is only a part of the total decision making process. However, since the method is very simple and well operationable, it can be linked very effectively with other parts in the total decision aid system, say prediction systems in portfolio selection and several analyzers in cement production. In particular, some of rules, which decide the way to do depending on the change of decision environment, can also be incorporated in the system as an artificial intelligence.

In portfolio problems such as bond trading, it is very important to get a solution reflecting faithfully the value-judgment of customers. In order to support bond traders in a flexible manner for the multiplicity of value-judgment and complex changes of economic environment, the cooperative system of man and computers are very attractive: above all, interactive multi-objective programming methods seem promising. We can say that interactive multi-objective programming makes the traditional mathematical programming flexible and robust (or 'sinayakana' in Japanese) to the environment of decision making such as uncertainty and the multiplicity of value judgment.

Acknowledgment

The author would like to express his sincere gratitude to Mr. N. Yasui of the Ube Kosan Co. Ltd. and people of the division of managial development of Cosmo Securities Co. Ltd. for their collaboration with this research. Also he is indebted to students of his laboratory, in particular, Mr. Sakamoto, for the development of computer programs. The computer experiments of this paper was carried out on Micro Vax 2000, and NEC PC-9800.

References

Borde, J. and Crouzeix, J. P. (1987). Convergence of a Dinkelbach-type Algorithm in Generalized Fractional Programming. Zeitschrift für Operations Research 31, pp. 31–54.

Ferland, A. and Potvin, J. (1985). Generalized Fractional Programming: Algorithms and Numerical Experimentation. *European J. Operational Research* 20, pp. 92–101.

Grauer, M., Lewandowski, A. and Wierzbicki A. P. (1984). DIDASS Theory, Implementation and Experiences. In: M. Grauer and A. P. Wierzbicki (eds.) Interactive Decision Analysis, Proceedings of an International Workshop on Interactive Decision Analysis and Interpretative Computer Intelligence, Springer, pp. 22–30.

Kito, N. and Misumi, M. (1978). An Application of Goal Programming to Cement Production. *Communications of the Operations Research Society of Japan* 23, pp. 177–181.

Konno, H. and Inori, M. (1987). Bond Portfolio Optimization by Bilinear Fractional Programming. Research Paper IKSS 87–4, Institute of Human and Social Sciences, Tokyo Institute of Technology.

Korhonen, P. and Wallenius, J. (1987). A Pareto Race. Working Paper F–180, Helsinki School of Economics.

Nakayama, H. (1984). Proposal of Satisficing Trade-off Method for Multiobjective Programming. *Transact. SICE*, 20, pp. 29–35. (in Japanese)

Nakayama, H. (1987). Sensitivity and Trade-off Analysis in Linear Multi-objective Programming. Presented in IIASA workshop in Albena/Bulgaria: also revised version is in preparation.

Nakayama, H. (1988). An Interactive Support System for Bond Trading. Presented at VIII-th International Conference on MCDM, Manchester.

Nakayama, H. and Sawaragi, Y. (1984). Satisficing Trade-off Method for Interactive Multiobjective Programming Methods. In: M. Grauer and A. P. Wierzbicki (eds.) Interactive Decision Analysis, Proceedings of an International Workshop on Interactive Decision Analysis and Interpretative Computer Intelligence. Springer, pp. 113–122.

Sawaragi, Y., Nakayama, H. and Tanino, T. (1985). Theory of Multiobjective Optimization, Academic Press.

Tversky, A. (1969). Intransitivity of Preferences. *Psychological Rev.*, 76, pp. 31–48.

On the Problem of Guaranteed Control with Vector-Valued Performance Criterion

O.I. Nikonov
Institute of Mathematics and Mechanics
Acad. Sci. USSR, Sverdlovsk, USSR

Abstract

The report is devoted to the problems closely related to those investigated in (Kurzhanski, 1986). For a dynamic system two varieties of restrictions depending upon the vector parameters μ and ν are given. The first one describes the phase constraints on the state variables, while the second is interpreted as the quality characteristic of the process. The two reciprocal problems arise then: to find the set of parameters μ of the phase constraints which ensure the given quality ν; and vice versa to specify the parameters ν guaranteed by the phase constraints with μ given in advance.

The report deals with the above sets and their "maximal" and "minimal" points description. A model example is given. The problems under consideration are motivated by environmental studies (see e.g. Konstantinov, 1983).

1 Problem formulation

We shall consider the following linear dynamic system

$$\begin{gathered} \dot{x} = A(t)x + f(t), \\ x \in \mathbf{R}^n, \qquad t_0 \leq t \leq \vartheta, \end{gathered} \tag{1.1}$$

where $A(t)$ is a given continuous matrix and the vector function $f(t)$ is fixed.

It is assumed that we have the system of constraints of the type

$$\begin{gathered} \varphi_i(x(t_i)) \leq \mu_i, \qquad i = 1, \ldots, N; \\ t_0 \leq t_1 \leq \ldots \leq t_N \leq \vartheta, \end{gathered} \tag{1.2}$$

with the functions $\varphi_i(\cdot) : \mathbf{R}^n \to \mathbf{R}^1$ being convex, $-\infty < \varphi_i(x) \to +\infty, \ \|x\| \to +\infty$. Thus the constraints (1.2) depend on the vector $\mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_N \end{pmatrix} \in \mathbf{R}^N$.

Let the performance criterion of the process be given by the values $g_j(x(\tau_j))$, $j = 1,\ldots,M$; $t_0 \le \tau_1 \le \ldots \le \tau_N \le \vartheta$, where the functions $g_j(\cdot) : \mathbf{R}^n \to \mathbf{R}^1$ have the properties similar to those of $\varphi_i(\cdot)$.

One can define the quality with the vector $\nu = \begin{pmatrix} \nu_1 \\ \vdots \\ \nu_M \end{pmatrix} \in \mathbf{R}^M$:

$$g_j(x(\tau_j)) \le \nu_j, \qquad j = 1,\ldots,M \tag{1.3}$$

(The given level ν is assumed to be guaranteed when the conditions (1.3) hold). So, we have two similar varieties of restrictions on the trajectories of the system (1.1). The first one is interpreted here as the phase constraints, while the second — as a characteristic of the quality.

The following reciprocal problems may be considered.

Problem 1 (inverse). With the quality $\nu \in \mathbf{R}^M$ given to specify the set $\mathcal{M}(\nu) \subseteq \mathbf{R}^N$, consisting of the vectors μ with the properties:

a) there exists a solution $x(\cdot)$ to (1.1) that (1.2) holds;

b) for every solution $x(\cdot)$ to (1.1)–(1.2) the relations (1.3) are true (the constraints guarantee the given quality ν).

Problem 1' (direct). For the given $\mu \in \mathbf{R}^N$ to specify the set $\mathcal{N}(\mu)$ of all the vectors $\nu \in \mathbf{R}^M$ such that for every solution $x(\cdot)$ to (1.1)–(1.2) the relations (1.3) are true.

Denote the set of Pareto maximal and weak Pareto maximal (Sawaragi et al., 1985) points of $\mathcal{M}(\nu)$ by $\mathcal{M}^P(\nu)$ and $\mathcal{M}^{WP}(\nu)$ respectively. (With respect to ordering introduced by $\mathbf{R}^N_+ = \{\mu \in \mathbf{R}^N : \mu_i \ge 0,\ i = 1,\ldots,N\}$. Analogously $\mathcal{N}^P(\mu)$ and $\mathcal{N}^{WP}(\mu)$ will stand for Pareto minimal and weak Pareto minimal elements of the set $\mathcal{N}(\mu)$.

Problem 2. Specify the sets $\mathcal{M}^P(\nu)$ and $\mathcal{M}^{WP}(\nu)$.

Problem 2'. Specify the sets $\mathcal{N}^P(\mu)$ and $\mathcal{N}^{WP}(\mu)$.

One has the simple situation when $M = N = 1$, $t_1 = t_0$, $\tau_1 = \vartheta$. In this case the set $\mathcal{M}(\nu)$ describes the level sets $L_\varphi(\mu) = \{x \in \mathbf{R}^n : \varphi(x) \le \mu\}$ containing the initial states of the system (1.1) from which the trajectories achieve the prescribed level set $L_g(\nu) = \{x \in \mathbf{R}^n : g(x) \le \nu\}$. The set $\mathcal{M}^P(\nu) = \mathcal{M}^{WP}(\nu)$ containing not more than one element characterises here the "maximal" domain of such initial states. Analogously the set $\mathcal{N}(\mu)$ consists of parameters ν for which the attainability domain from the set $L_\varphi(\mu)$ is contained in the $L_g(\nu)$, and the element $\nu^0 : \mathcal{N}^P(\mu) = \{\nu^0\}$ (with $L_\varphi(\mu) \ne \emptyset$) is the minimal one with such a property.

2 Basic relations

We shall further use the following notations:

Denote

$$L = [l^{(1)}, \dots, l^{(N)}] \in \mathbf{R}^{n \times N}, \qquad \varphi^*(L) = \begin{pmatrix} \varphi_1^*(l^{(1)}) \\ \vdots \\ \varphi_N^*(l^{(N)}), \end{pmatrix},$$

$$D = [d^{(1)}, \dots, d^{(M)}] \in \mathbf{R}^{n \times M}, \qquad g^*(D) = \begin{pmatrix} g_1^*(d^{(1)}) \\ \cdots \\ g_M^*(d^{(M)}), \end{pmatrix},$$

where $\varphi_i^*(l^{(i)})$, $g_j^*(d^{(j)})$ are the Fenchel conjugate to $\varphi_i(\cdot)$, $g_j(\cdot)$ $(\varphi_i^*(l^{(i)}) = \sup_{x \in \mathbf{R}^n}\{ (l^{(i)}, x) - \varphi_i^*(x) \})$; $\alpha = (\alpha_1, \dots, \alpha_N) \in \mathbf{R}_+^N$. Let $s_\varphi(t; \alpha, L)$ stands for the solution to the system

$$\dot{s} = -A^T(t)s + \sum_{i=1}^{N} \alpha_i \delta(t - t_i) l^{(i)}, \qquad t \le t_N, \tag{2.1}$$

$$s(t) \equiv 0, \qquad t > t_N \tag{2.2}$$

Here $\delta(t - t_i)$ are delta functions and the solution to (2.1)–(2.2) is treated in the sense of the theory of distributions.

Lemma 2.1. Let $\mu \in \mathbf{R}^N$ is given. Then for the solution $x(\cdot)$ to (1.1) the conditions (1.2) are fulfilled iff $\forall q \in \mathbf{R}^N$

$$\begin{gathered} (q, x(t_0)) \le \inf_{\alpha, L}\Big\{(\alpha, \mu + \varphi^*(L)) - \int_{t_0}^{\vartheta} s_\varphi^T(\tau; \alpha, L) f(\tau)\, d\tau\Big\} \\ s(t_0; \alpha, L) = q \end{gathered} \tag{2.4}$$

Denote by $s_g(t; \beta, D)$ a solution to the system (similar to (2.1)–(2.2))

$$\begin{gathered} \dot{s} = -A^T(t)s + \sum_{j=1}^{M} \beta_j \delta(t - \tau_j) d^{(j)}, \qquad t \le \tau_M, \\ s(t) \equiv 0, \qquad t > \tau_M. \end{gathered} \tag{2.5}$$

and introduce the function

$$\psi(\mu,\nu) = \sup_{\beta,D}\inf_{\alpha,L}\Big\{(\alpha,\mu+\varphi^*(L)) - (\beta,\nu+g^*(D)) + \\ + \int_{t_0}^{\vartheta}[s_\varphi(\tau;\beta,L) - s_g(\tau;\beta,D)]^T f(\tau)\,d\tau\Big\} \tag{2.6}$$

$$\alpha, L, \beta, D \ :\ s_\varphi(t_0;\alpha,L) = s_g(t_0;\beta,D).$$

Theorem 2.1. The sets $\mathcal{M}(\nu')$ and $\mathcal{N}(\mu')$ can be determined as the sets of all the solutions to the inequalities

$$-\infty < \psi(\mu,\nu') \le 0 \tag{2.7}$$

and

$$\psi(\mu',\nu) \le 0 \tag{2.8}$$

respectively.

Remark. Under conditions on the functions $\varphi_i(\cdot)$, $g_j(\cdot)$ given in section 2 one always has $\mathcal{N}(\mu) \neq \emptyset$, while $\mathcal{M}(\nu)$ can be empty. The left part of (2.7) ensures that condition a) of the problem 1 is fulfilled.

The further structural properties of the sets in question are given in the following assertion.

Lemma 2.2.

i) The sets $\mathcal{M}(\nu)$ are closed and have the following convexity property:

$$\mu^{(1)},\mu^{(2)} \in \mathcal{M}(\nu),\ \ \mu^{(2)} - \mu^{(1)} \in \mathbf{R}^N_+ \implies \forall \mu : \mu_i^{(1)} \le \mu_i \le \mu_i^{(2)},\ \ \mu \in \mathcal{M}(\nu).$$

ii) If for the given $\mu \in \mathbf{R}^N$ condition a) of the problem 1 holds than $\mathcal{N}^P(\mu)$ is a singleton $\{\nu^0\} = \mathcal{N}^P(\mu)$ and $\mathcal{N}(\mu) = \nu^0 + \mathbf{R}^M_+$; otherwise $\mathcal{N}(\mu) = \mathbf{R}^M$.

To specify the sets $\mathcal{M}^P(\nu)$, $\mathcal{M}^{WP}(\nu)$ the following functions may be considered.

Denote

$$\begin{gathered} \mu \setminus i = (\mu_1,\dots,\mu_{i-1},\mu_{i+1},\dots,\mu_N)^T \qquad (\mu \in \mathbf{R}^N), \\ \omega_i(\mu\setminus i,\nu) = \sup\{\,\mu_i' \mid -\infty < \psi(\mu',\nu) \le 0,\ \mu_k' = \mu_k,\ k \neq i\,\} \end{gathered} \tag{2.9}$$

$\omega_i : \mathbf{R}^N \to [-\infty,+\infty]$ ($\omega_i(\mu\setminus i,\nu) = -\infty$ iff there is no μ_i' among which sup in (2.9) is sought).

Let then $\Omega_i(\nu)$ be the graphs of the function ω_i: $\Omega_i(\nu) = \{\,\mu \in \mathbf{R}^N : \mu_i = \omega_i(\mu\setminus i,\nu)\,\}$ and

$$\nu_j^0 = \sup_{d^{(j)}}\inf_{\alpha,L}\Big\{(\alpha,\mu+\varphi^*(L)) - g_j^*(d^{(j)}) + \int_{t_0}^{\vartheta}[s_\varphi(\tau;\alpha,L) - s^0(\tau,d^{(j)})]^T f(\tau)\,d\tau\Big\} \tag{2.10}$$

where $s^0(\tau, d^{(j)}) = s_g(\tau; e^{(j)}, D)$, $e^{(j)} = (0, \ldots, 1, \ldots, 0)^T$.

Theorem 2.2.

i) The following equalities are true:

$$\mathcal{M}^{WP}(\nu) = cl \bigcup_{i=1}^{N} \Omega_i(\nu); \qquad \mathcal{M}^P(\nu) = \bigcap_{i=1}^{N} \Omega_i(\nu)$$

ii) If ν_j^0, $j = 1, \ldots, M$, defined in (2.10) are finite, then $\mathcal{N}^P(\mu) = \{\nu^0\}$, otherwise $\mathcal{N}^P(\mu) = \emptyset$.

3 Example

Consider the system

$$\begin{aligned} & & \dot{x}_1 &= x_2 + f_1(t), & & \\ & & \dot{x}_2 &= -x_1 + f_2(t), & & \\ N = 2, & & \varphi_i(x) &= (x - x_0^{(i)})^T A_i (x - x_0^{(i)}), & A_i > 0, & \quad i = 1, 2, \\ M = 1, & & g(x) &= (x - x_g)^T B (x - x_g), & B > 0 & \end{aligned}$$

$$t_0 \le t_1 \le t_2 \le \vartheta; \qquad t_0 \le \tau_1 \le \vartheta.$$

Relations (2.7)–(2.8) can be here interpreted as follows.
Denote

$$\begin{aligned} \tilde{\varphi}_i(x) &= (x - \tilde{x}_0^{(i)})^T \tilde{A}_i (x - \tilde{x}_0^{(i)}), \\ \tilde{g}(x) &= (x - \tilde{x}_g)^T \tilde{B} (x - \tilde{x}_g), \end{aligned}$$

where

$$\begin{aligned} \tilde{x}_0^{(i)} &= X(t_0, t_i) x_0^{(i)} - \int_{t_0}^{t_i} X(t_0, \tau) f(\tau)\, d\tau, & \tilde{A}_i &= X(t_0, t_i) A_i X(t_i, t_0), \\ \tilde{x}_g &= X(t_0, \tau_1) x_g - \int_{t_0}^{\tau_1} X(t_0, \tau) f(\tau)\, d\tau, & \tilde{B} &= X(t_0, \tau_1) B X(\tau_1, t_0), \end{aligned}$$

$$X(\vartheta, t) = \begin{pmatrix} \cos(\vartheta - t) & \sin(\vartheta - t) \\ -\sin(\vartheta - t) & \cos(\vartheta - t) \end{pmatrix}$$

and

$$\begin{aligned} \varepsilon_i(\mu_i) &= \{\, x \in \mathbf{R}^n;\ \tilde{\varphi}_i(x) \le \mu_i \,\}, \\ \varepsilon(\nu) &= \{\, x \in \mathbf{R}^n : \tilde{g}(x) \le \nu \,\}. \end{aligned}$$

Then we have $\mu \in \mathcal{M}(\nu)$ iff $\emptyset \neq \varepsilon_1(\mu) \cap \varepsilon_2(\mu_2) \subseteq \varepsilon(\nu)$. Some computational results are shown in Fig. 1–2. The first one shows the set $\mathcal{M}(\nu)$ for the following values of parameters:

$$\tilde{x}_0^{(1)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \tilde{x}_0^{(2)} = \begin{pmatrix} 4 \\ -3 \end{pmatrix}, \quad \tilde{x}_g = \begin{pmatrix} 1.5 \\ -2.3 \end{pmatrix},$$

$$\tilde{A}_1 = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}, \quad \tilde{A}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 8 & 0 \\ 0 & 1 \end{pmatrix}, \quad \nu = 9.$$

The configuration of the sets ε_1, ε_2, ε in this case for $\tilde{\mu}_1 = \tilde{\mu}_2 = 10.71$ is shown in Fig. 2.

Conclusions

For a class of linear dynamic systems with constraints depending upon the vector-valued parameters two reciprocal problems concerning the sets of the "admissible" parameters description are discussed. A model example is given. Further applications of the presented technique may be done for uncertain dynamic systems using the methods developed in (Kurzhanski, 1977; Kurzhanski and Filippova, 1986; Kurzhanski and Nikonov, 1984).

References

Konstantinov, G.N. (1983). The normalizing of the actions on dynamic system. Irkutsk.

Krasovskii, N.N. (1985). The control of a dynamic system. Moscow, Nauka.

Kurzhanski, A.B. (1977). Control and estimation under uncertainty. Moscow, Nauka.

Kurzhanski, A.B. (1986). Inverse problems in multiobjective dynamic optimization. Proc. VII Intern. Conf. on Multiple Criteria Decision Making, Kyoto, 1986. *Lecture Notes Econ. & Math. Syst.*, Vol. 285.

Kurzhanski, A.B. and Filippova, T.F. (1986). Valable solutions for uncertain systems. IIASA. CP-86-11.

Kurzhanski, A.B. and Nikonov, O.I. (1984). On the problem of guaranteed control for uncertain systems. Proc. IX Congr. IFAC, Budapest.

Sawaragi, Y., Nakayama, H. and Tanino, T. (1985). Theory of multiobjective optimization. New York, Acad. Press.

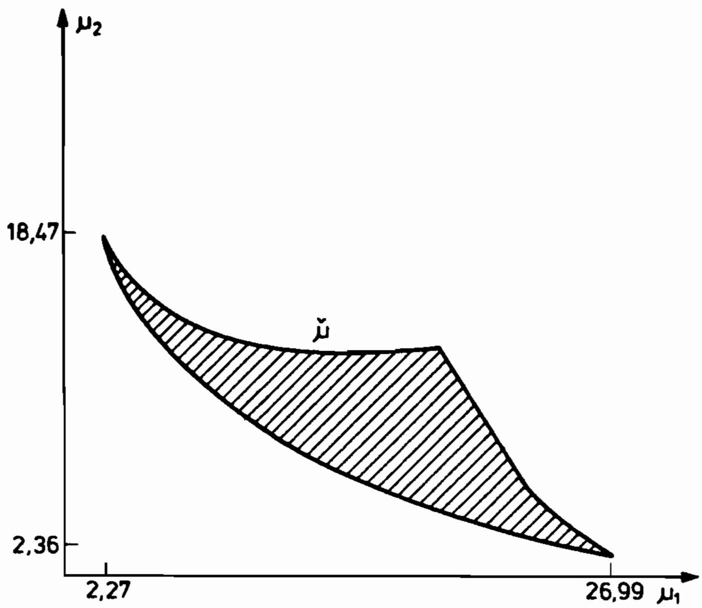


Figure 1

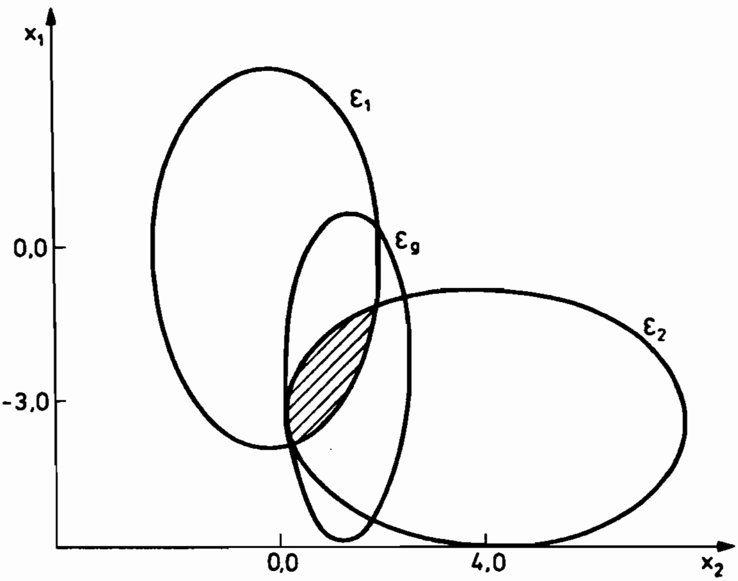


Figure 2

Applying Multi-Criteria Optimization for Effective Combinatorial Problem Solving

B. V. Peltzverger, O. V. Havronin
Chelyabinsk Polytechnical Institute
Chelyabinsk, USSR

Introduction

It is shown that there exists a non-empty class of one-criterion combinatorial problems which can be solved more effectively by introducing particular criteria and solving the problem as a multi-criteria one (decomposition approach (Krasnoschokov, et al., 1979).

A class of optimization combinatorial problems (D-class) for which the equivalent recognition problems are NP-complete or open is introduced. D-class is such that according to decomposition approach there exists particular criteria compatible with the main one , and a recognition task corresponding to a particular criterion always belongs to P-class. It is shown that a class thus introduced is not empty.

Decomposition approach allows to construct approximate polynomial algorithms for D-class problems. Indeed, to determine a non-dominated point from an admissible set we must solve a recognition problem for every particular criterion. By finding a necessary number of non-dominated points we construct approximately Pareto's set where according to decomposition approach there is a solution of the initial problem. As a matter of fact, searching a solution on Pareto's set is not required as for every non-dominated point the main criterion is calculated and only its minimum (maximum) value survives.

The approach presented here was used to solve a transport networks synthesis and coordination problems.

1 Definition a class D

Let's consider initial NP-hard combinatorial optimization problems (IP)

$$F(x) \to \min, \qquad x \in X, \tag{1.1}$$

where X — is a finite set.

Along with the main criterion for the problems in question it is possible to introduce monotonically coordinated particular criteria $u_i(x) \to \min$, $i = \overline{1,m}$. Particular criteria are monotonically coordinated if for any $x', x'' \in X$, $F(x') \leq F(x'')$ follows $u_i(x') \leq u_i(x'')$, $\forall i \in \overline{1,m}$.

The problem

$$u_i(x) \to \min, \qquad x \in X \tag{1.2}$$

is a particular problem (PP).

Definition. A class D is NP-hard problems (1.1) for which there exist monotonically coordinated particular criteria and a particular problems (1.2) belongs to a class P.

Let's show that a class thus introduced is not empty.

Problem 1. Optimal communication spanning tree (Garey and Johnson, 1979).

Given is a complete graph $G = (V, E)$, weight $w_{ij} > 0$, $\forall (i,j) \in E$ requirement z_{ij} for all pair vertices $i, j \in V$.

$$\text{IP}: \qquad F(T) = \sum_{i \in V} \sum_{j \in V} W_{ij}(T) \cdot z_{ij} \to \min, \quad T \in \Omega,$$

where $W_{ij}(T)$ is the path length in T from i to j, Ω is the set of the all spanning trees of the graph G.

$$\text{PP}_i: \qquad u_i(T) = \sum W_{ij}(T) z_{ij} \to \min, \qquad T \in \Omega, \quad i = \overline{1,n}$$

where $n = |V|$. PP_i is a shortest path tree problem, belongs to the class P. The condition of coordination is satisfied, as $F(T) = \sum_{i=1}^{n} u_i(T)$.

Problem 2. Transport network synthesis (Podinovsky and Nogin, 1982).

Given is a graph $G = (V, E)$ with a singled out vertex $i_0(\sin k)$, $\forall (i,j) \in E$ given are arc laying cost $w_{ij} \geq 0$, a unit product transportation cost along an arc $v_{ij} \geq 0$, $\forall i \in V \setminus \{i_0\}$ given is the power of a vertex $P_i \geq 0$.

$$\text{IP}: \qquad F(T) = \sum_{(i,j) \in T} (w_{ij} + v_{ij} y_{ij}) \to \min, \qquad T \in \Omega,$$

where T – is a spanning tree of graph G, Ω is a set of the spanning trees of the graph G, y_{ij} is a flow along an arc in T.

The NP-hardness of this problem is proved by reducing a minimal covering problem to it.

$$\text{PP}_1: \ u_1(T) = \sum_{(i,j) \in T} w_{ij} \to \min, \quad T \in \Omega \ -$$ a minimal spanning tree problem, belongs to the class P.

$$\text{PP}_2: \ u_2(T) = \sum_{(i,j) \in T} v_{ij} y_{ij} \to \min, \quad T \in \Omega \ -$$ a shortest path tree problem, belongs to the class P.

The condition of coordination is satisfied, as $F(T) = u_1(T) + u_2(T)$

Problem 3. Transport networks coordination (Podinovsky and Nogin, 1982).

Given is a graph $G = (V, E)$, $V = I_1 \cup I_2 \cup \{i_0\}$, $E = E_1 \cup E_2$, where $E_k = \{(i,j) \mid i, j \in I_k \cup \{i_0\}\}$, $k = \overline{1,2}$, $E_1 \cap E_2 \neq \emptyset$, $\forall (i,j) \in E_k$ given are arc laying and k-th product transportation costs $w_{ij}^{(k)} \geq 0$, $v_{ij}^{(k)} \geq 0$, $k = \overline{1,2}$. $\forall (i,j) \in E_1 \cap E_2$ a cost of the common arc laying is given $w_{ij} \geq 0$ and such condition is satisfied:

$$\max \{w_{ij}^{(1)}, w_{ij}^{(2)}\} \leq w_{ij} \leq w_{ij}^{(1)} + w_{ij}^{(2)}.$$

$\forall i \in I_k$ a power on the k-th product $P_i^{(k)} \geq 0$, $k = \overline{1,2}$ is given.

$$\text{IP} \quad : \quad F(T_1,T_2) = \sum_{(i,j)\in T_1\setminus T_2} \left(w_{ij}^{(1)} + v_{ij}^{(1)}y_{ij}^{(1)}\right) + \sum_{(i,j)\in T_2\setminus T_1} \left(w_{ij}^{(2)} + v_{ij}^{(2)}y_{ij}^{(2)}\right) +$$

$$+ \sum_{(i,j)\in T_1\cap T_2} \left(w_{ij} + v_{ij}^{(1)}y_{ij}^{(1)} + v_{ij}^{(2)}y_{ij}^{(2)}\right) \to \min, \quad (T_1,T_2) \in \Omega_1 \times \Omega_2$$

where Ω_k is the set of the spanning trees of the graph $G_k = (I_k \cup \{i_0\}, E_k)$, $k = \overline{1,2}$.

The problem is NP-hard as it is a generalization of problem 2. It is possible to introduce the coordinated particular criteria this way:

$$F(T_1,T_2) = F_1(T_1) + F_2(T_2) + W(T_1,T_2)$$

where

$$F_k(T_k) \;=\; \sum_{(i,j)\in T_k} \left(w_{ij}^{(k)} + v_{ij}^{(k)}y_{ij}^{(k)}\right), \quad k = \overline{1,2}.$$

$$W(T_1,T_2) \;=\; \sum_{(i,j)\in T_1\cap T_2} \left(w_{ij} - w_{ij}^{(1)} - w_{ij}^{(2)}\right).$$

PP with criterion $W(T_1,T_2)$ is a maximal spanning tree problem, belongs to the class P.

PP $F_k(T_k) \to \min$, $T_k \in \Omega_k$, $k = \overline{1,2}$, is a problem of transport network synthesis (Problem 2) and may be divided into a minimal spanning tree problem and a shortest path tree problem.

2 A general scheme of solving problems from a class D

According to a decomposition approach (Krasnoschokov, et al., 1979) the global extremum of problem (1.1) is reached on the effective solutions set of the problem

$$u(x) = \left(u_1(x), \ldots, u_m(x)\right) \to \min, \qquad x \in X. \tag{2.1}$$

Let P_u, P_x - effective solutions sets in the space of criteria and the space of solutions correspondingly. So the problem (1.1) is reduced to the problem

$$F(x) \to \min, \qquad x \in P_x.$$

It's evident that such an approach can be realized if the construction of the set $P_u(P_x)$ is not too labour consuming. Let $\Lambda = \{\lambda \in R^m \mid \sum_{i=1} \lambda_i = 1, \lambda_1 > 0, \forall i = \overline{1,m}\}$. Let's consider the problem

$$\max_{i\in\overline{1,m}} \lambda_i u_i(x) \to \min, \qquad x \in X \tag{2.2}$$

According to the Germeyer theorem (Podinovsky and Nogin, 1982), if the set of estimates is bounded, closed and is entirely inside the orthant $R^m_{\geq}$, $P_u(P_x)$ can be constructed as a result of solving problem (2.2) for all $\lambda \in \Lambda$ as the set of estimates is finite.

In the class D one can find two subclasses:

1. Problem (2.2) belongs to the class P. Let D_1 be a subclass of problems from D, for which this statement is true.

2. Problem (2.2) is NP-hard or open. Let D_2 be a subclass of D, for which this statement is true.

For problem from the subclass D_1 when construction the set of effective solutions traditional methods based on convolution of particular criteria of the type (2.2) can be used (Podinovsky and Nogin, 1982).

For the problems of the subclass D_2 this way is not suitable as for the construction of each effective point it is necessary to solve a problem as complex as the initial problem. Problems 1–3 belong to this class.

To solve problems from D_2 a multipass algorithm based on the construction of sets $\tilde{P}_u^{(k)}(\tilde{P}_x^{(k)})$ which are the corresponding approximations of the sets $P_u(P_x)$ on the k-th path is proposed. The number of effective points does not decrease from path to path which because of the finiteness of the admissible set X guarantees the construction of $P_u(P_x)$. The sets $\tilde{P}_u^{(k)}(\tilde{P}_x^{(k)})$ are used for obtaining an approximate solution of the problem by memorizing the record value $F(x)$, $x \in \tilde{P}_x^{(k)}$ and besides, they allow to construct an upper bound on each path.

Let the convex hull of the projection of the admissible set in the space of particular criteria be

$$U = \text{conv } \{u(x) \mid x \in X\}.$$

On the zero path from the admissible set X and, simultaneously, from U a subset $CP_u^{(0)}(CP_x^{(0)})$ is extracted such that $P_u \subset CP_u^{(0)} \subset U$ $(P_x \subset CP_x^{(0)} \subset X)$.

$CP_u^{(0)}$ and $CP_x^{(0)}$ are constructed as follows:

$$\begin{array}{lll} \text{for all } i = \overline{1,m} & u_i^{(i)} = \min\limits_{x \in X} u_i(x), & x^{(i)} = arg \min\limits_{x \in X} u_i(x) \\ & \forall j \neq i & u_j^{(i)} = u_j(x^{(i)}) \text{ are determined} \end{array}$$

Let $\underline{u}_i = u_i^{(i)}$, $\overline{u}_i = \max_{j \in \overline{1,m}} u_i^{(j)}$ then $CP_u^{(0)} = [\underline{u}_1, \overline{u}_1] \times \ldots \times [\underline{u}_m, \overline{u}_m]$, $CP_x^{(0)} = \{x \in X \mid u(x) \in CP_u^{(0)}\}$.

So, solving particular problems of polynomial labour consumption allows to reduce the initial admissible set by omitting ineffective points which can't be the solution of problem (1.1)

$$\begin{aligned} \tilde{P}_x^{(0)} &= \{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}, \\ \tilde{P}_u^{(0)} &= \{u(x^{(1)}), u(x^{(2)}), \ldots, u(x^{(m)})\} \end{aligned}$$

Not to select the points from X again (to avoid looping) a set $\widetilde{CP}_x^{(k)}$, $\widetilde{CP}_x^{(0)} = CP_x^{(0)} \setminus \tilde{P}_x^{(0)}$ is introduced.

Suppose that there exist a subdivision of the admissible set X into subsets $X_i^{(k)}$, $i = \overline{1, N^{(k)}}$, $k = \overline{1, L}$ (where L is the number of paths of the algorithm, $N^{(k)}$ is the number of subsets on the k-th path) such that $|X_i^{(k)}| \approx |X|/N^{(k)}$, $X = \bigcup_{k=1}^{L} \bigcup_{i=1}^{N^{(k)}} X_i^{(k)}$, $\forall\, i, j, i \neq j$, $\forall\, k \in \overline{1, L}$, $X_i^{(k)} \cap X_j^{(k)} = \emptyset$. Labour consumption of $x_i^{(k)} \in X_i^{(k)}$ generation is bounded by the polynomial of the measure of initial problem (1.1). In particular, if the subdivision

X is given explicitly, the labour consumption of $x_i^{(k)}$ generation is constant and does not depend on the measure of the measure of the problem. If additional conditions are not imposed on the subdivision X then the algorithm of $\tilde{P}_u^{(L)}$ construction is analogous to (Sobol and Statnikov, 1981) and is as follows:

Step 1 $k = 1$

Step 2 $CP_u^{(k)} = CP_u^{(k-1)}$, $CP_x^{(k)} = CP_x^{(k-1)}$, $\tilde{P}_u^{(k)} = \tilde{P}_u^{(k-1)}$, $\tilde{P}_x^{(k)} = \tilde{P}_x^{(k-1)}$, $\widetilde{CP}_x^{(k)} = \widetilde{CP}_x^{(k-1)}$, $i = 1$.

Step 3 $x_i^{(k)} \in X_i^{(k)} \cap \widetilde{CP}_x^{(k)}$ is chosen. The effectiveness of $x_i^{(k)}$ in $\tilde{P}_x^{(k)}$ is checked. If effective: $\tilde{P}_x^{(k)} = \tilde{P}_x^{(k)} \cup \{x_i^{(k)}\}$, $\tilde{P}_u^{(k)} \cup \{u(x_i^{(k)})\}$, $CP_u^{(k)} = CP_u^{(k)} \setminus \mathcal{D}(x_i^{(k)})$, where $\mathcal{D}(x_i^{(k)}) = \{u(x) \mid x \in CP_x^{(k)}, u(x_i^{(k)}) \leq u(x)\}$, $CP_x^{(k)} = \{x \in CP_x^{(k)} \mid u(x) \in CP_u^{(k)}\}$, $\tilde{P}_u^{(k)} = \tilde{P}_u^{(k)} \setminus \mathcal{D}(x_i^{(k)})$, $\tilde{P}_x^{(k)} = \{x \in \tilde{P}_x^{(k)} \mid u(x) \in \tilde{P}_u^{(k)}\}$

Step 4 $\widetilde{CP}_x^{(k)} = \widetilde{CP}_x^{(k)} \setminus \{x_i^{(k)}\}$, $i = i + 1$ if $i \leq N^{(k)}$ go to step 3, otherwise $k = k + 1$ and if $k \leq L$ go to step 2, otherwise — end.

So on the k-th path sets $CP_u^{(k)}(CP_x^{(k)})$ and $\tilde{P}_u^{(k)}(\tilde{P}_x^{(k)})$ are constructed and for any k $CP_U^{(k)} \supset P_u$, $CP_x^{(k)} \supset P_x$, $CP_u^{(k)} \subseteq CP_u^{(k-1)}$, $CP_x^{(k)} \subseteq CP_x^{(k-1)}$, $(\tilde{P}_u^{(k)} \cap P_u) \supseteq (\tilde{P}_u^{(k-1)} \cap P_u)$. Because of the finiteness of X there exist $L = L^*$ such that $CP_u^{(L^*)} \cap \{u(x) \mid x \in X\} = P_u$, $CP_x^{(L^*)} = P_x$.

Denote $F_k = \min_{x \in \tilde{P}_x^{(k)}} F(x)$, $k = \overline{0, L}$. According to the scheme described we have $\min_{x \in X} F(x) \leq F_L \leq \ldots \leq F_0$, when $L = L^*$, $\min_{x \in X} F(x) = F_{L^*}$.

For real problems the value of L^* turns out to be inadmissibly large and the value of L is to be chosen according to the resources available. The experience of the authors shows that a small number of paths often allows to obtain a satisfactory result. If there exists a functional dependence of the type $F(x) = \Phi(u_1(x), \ldots, u_m(x))$ then the algorithm allows to construct a lower bound as well on the basis of particular problems solutions:

$$\Phi(u_1^{(1)}, \ldots, u_m^{(m)}) \leq \min_{x \in X} F(x)$$

For the problems in question it is possible to construct the subvision X which is monotonical in one of the particular criteria u_z:

$$\forall k, \quad \forall i, j, \quad i < j, \quad \max_{x \in X_i^{(k)}} u_z(x) \leq \min_{x \in X_j^{(k)}} u_z(x).$$

This permits to make a rapid and uniform pass along one of the axis of the space of criteria.

3 Conclusions

A method of solving problems 2 and 3 based on the general scheme given above. As the initial criterion is the sum of the particular ones the solution of problem can be obtained when constructing a set of effective solutions P of the problem (2.1) by memorizing a record value $F(x)$. The subdivision admissible set Ω in problem 2 is based on the generator spanning trees in order of weight increasing (Gabow, 1977). The generator is modified such as the spanning trees are generated in the order of weight increasing but in any previously determined step. This permits to make a uniform sample in space of particular criteria. The subdivisions admissible set in problem 3 is based on the generation the forests of the graph $G_0 = (I_1 \cap I_2 \cup \{i_0\}, E_1 \cap E_2)$ in order of weight decreasing.

Implementation has shown high effectiveness of the approach considered for the transport networks coordination problem.

References

Gabow, H. N. (1977). Two algorithms for generating weighted spanning trees in order. *SIAM J. Comput.*, Vol. 6, No 1.

Garey, M., Johnson, D. (1979). Computers and intractibility. San Francisco: Freeman and Company.

Krasnoschokov, P. S., Morosov, V. V., Fyodorov, V. V. (1979). Decomposition approach to designing problems. Izvestia AS USSR. *Technicheskaya Kibernetica*, No 2. (in Russian)

Peltzverger, B. V., Havronin, O. V. (1988). Applying of decomposition approach for complex combinatorial problem solving. Izvestia AS USSR. *Technicheskaya Kibernetica*, No 3. (in Russian)

Podinovsky, V. V., Nogin, V. D. (1982). Pareto-optimal solutions of multiple-criteria problems. Nauka, (in Russian).

Sobol, I. M., Statnicov, R. B. (1981). Optimal parameters selection for multiple-criteria problems. Nauka, (in Russian).

Criteria Importance Theory

V.V. Podinovskii
USSR Union Movement School
Moscow, USSR

The majority of known methods of multicriterial optimization presupposes the application of information of the importance of the criteria in one or another form. But at the same time the concepts of ordering of criteria according to their importance are being introduced in very different ways in connection with models used. And sometimes the definitions of criteria importance are incorrect or they are absent at all (see a survey in Podinovskii, 1979).

Therefore the problem of creating a rigorous and constructive theory of criteria importance happened to be topical. Such theory for the basic forms of importance has been elaborated in (Podinovskii, 1975, 1976, 1978a, 1978b, 1978c, 1979; Menshikova and Podinovskii, 1978; Podinovskii and Polishchuck, 1988). It is axiomatic and bases on precise definitions of comparison by importance of single criteria and sets of criteria, and also of comparisons of degrees of superiority in importance. These definitions are simple and can be used during an interrogation of the decision-maker for obtaining information on his preferences.

A careful analysis shows that the notion of *a criteria importance* is a representation of a special kind of regularity of a preference structure. This regularity provides increase or stability of preferences under specific ratio of increases of those components of vector estimates which correspond to the criteria comparing by importance (with all other components fixed).

We introduce general definitions of the notion of ordering of criteria by importance in connection with the multicriterial optimization problem under certainty. In this problem there are a set of available alternative decisions (strategies) S and a vector criterion $f = (f_1, \dots, f_m)$ defined on S. Each strategy s is represented by its vector estimate $f(s) \in X$, where $X = X_1 \times \dots \times X_m \subseteq Re^m$ is a set of all vector estimates.

The DM's preferences are being represented by a non-strict preference relation R^* from a class $\mathcal{R}$ of binary relations on X. The class $\mathcal{R}$ is defined by suitable assumptions ("normative"specifications). We'll consider the next two classes:

$\mathcal{R}_q$ — a set of partial quasiorders (i.e. reflexive and transitive relations);

$\mathcal{R}_{qc}$ — a set of partial convex quasiorders (i.e. the quasiorders on a convex set X, for which the set $R(x) = \{ y \in X \mid y\, R\, x \}$ is convex for each $x \in X$).

We shall denote by I and P respectively the indifference and (strict) preference relations corresponding to $R : I = R \cap R^{-1};\ \ P = R \setminus I$.

The relation R^* is unknown and should be restored (completely or partially) by the information on DM's preferences. Such information may be one on the relative importance of criteria Ω consisting of single judgments ω of the criteria importance.

We'll consider the multicriterial maximization problem for which the Pareto relation R^0 is defined as follows: $x\, R^0\, y$ iff$x_i \geq y_i$, $i = 1, \ldots, m$. Being a part of the relation R^* it is embedded into $R^* : R^0 \leq R^*$, i.e. $I^0 \subseteq I^*$ and $P^0 \subseteq P^*$ are both hold.

According to the definitions given below every judgment ω about criteria importance induces an indifference relation I^ω or a preference relation P^ω on X. Besides the importance information $\Omega = \{\omega_1, \ldots, \omega_k\}$ let us have the information Θ which induces a relation R^Θ on X. Then using all the information $\Theta \cup \Omega$we can introduce on X a relation

$$R^{\Theta\cup\Omega} = Cl_{\mathcal{R}}[R^\Theta \cup R^{\omega_1} \cup \ldots \cup R^{\omega_k}],$$

where by $Cl_{\mathcal{R}} R$ we denote a closure of R in $\mathcal{R}$, i.e. an intersection of all $\hat{R} \in \mathcal{R}$ such that $R \leq \hat{R}$. Further we shall consider the case when $R^\Theta = R^0$ and write R^Ω instead of $R^{\Theta\cup\Omega}$.

Definition N. The information Ω is said to be *consistent* (in $\mathcal{R}$) iff $R^{\omega_j} \leq R^\Omega$, $j = 1, \ldots, k$, and $R^0 \leq R^\Omega$.

Now let us give the basic definitions of the criteria importance, restricting ourselves here to a bicriterial problem ($m = 2$) for simplicity. Lower the criteria will be denoted by their numbers (1 or 2).

In each definition there are vector estimates x, y of a special mode from a set X^1. The pare x, y is formed on a basic vector estimate z from a set X^0 as follows:

$$x = (h_1(z_1), z_2), \qquad y = (z_1, h_2(z_2)), \tag{1}$$

where h_i ($i = 1, 2$) is a numerical function of "positive shift"defined on $\widehat{X_i} \supseteq X_i$, i.e. $h_i(t) > t$ for all t. A set of the vector-functions $h = (h_1, h_2)$ which can be used in (1) will be denoted by H.

Definition $I^\sim$. Assertion $\omega^\sim = \langle$ criteria 1 and 2 are of *equal importance* (from X^0 to X^1 according to H) $\rangle$ means that for any $z \in X^0$ and $h \in X$ every two vector estimates x, y formed by (1) and lying in X^1 are indifferent ($x\, I^{\omega^\sim}\, y$ and $y\, I^{\omega^\sim}\, x$).

Definition $P^\succ$. Assertion $\omega^\succ = \langle$ the criterion 1 is *more important* than the criterion 2 (from X^0 to X^1 according to H) $\rangle$ means that of every two vector estimates x, y formed by (1) and lying in X^1 the first estimate is more preferable than the second one ($x\, P^{\omega^\succ}\, y$).

According to the given definitions the criteria importance can be called free when $X^0 = X$ and bound (to X^0) when $X^0 \neq X$, and also global when $X^1 = X$ and local when $X^1 \neq X$.

Contributing a subscript ω to the sets X^0 and X^1 and an upscript to the set H from the definition of ω we shall introduce designations:

$$X^0_\Omega = \bigcap_{\omega\in\Omega} X^0_\omega, \qquad X^1_\Omega = \bigcap_{\omega\in\Omega} X^1_\omega.$$

By $\geq^{\omega}$ we'll mean $=$ when $\omega = \omega^{\sim}$ and $>$ when $\omega = \omega^{\succ}$.

Definition C. The positive numbers λ_1, λ_2 which assure the compatibility of a system

$$\lambda_1(x_1 - y_1) \geq^{\omega} \lambda_2(y_2 - x_2), \qquad \omega \in \Omega \tag{2}$$

for any $z \in X^0_{\Omega}$, $h^{\omega} = H^{\omega}$ and $x, y \in X^1_{\Omega}$ from (1) are called the *coefficients of importance of criteria* generated by Ω.

Having rewrited (2) using (1):

$$\lambda_1(h^{\omega}_1(z_1) - z_1) \geq^{\omega} \lambda_2(h^{\omega}_2(z_2) - z_2), \qquad \omega \in \Omega \tag{3}$$

we see that after the criteria scales have been changed by means of coefficients of importance, the increases of criteria of equal importance become equal, and the increase of the more important criteria occurs to be greater than of the less important one.

Denote by $\Lambda^{\Omega} \subset Re^2_+$ the set of the coefficients of importance generated by Ω. The other form of (2)

$$\lambda_1 x_1 + \lambda_2 x_2 \geq^{\omega} \lambda_1 y_1 + \lambda_2 y_2, \qquad \omega \in \Omega \tag{4}$$

shows that a linear function $L(x|y) = \lambda_1 x_1 + \lambda_2 x_2$ with $\lambda \in \Lambda^{\Omega}$ provides "linearization" of the relation R^{Ω}, semi-representing the relations I^{ω} and P^{ω}, $\omega \in \Omega$:

$$x\, I^{\omega}\, y \implies L(x|\lambda) = L(y|\lambda); \qquad x\, P^{\omega}\, y \implies L(x|\lambda) > L(y|\lambda).$$

It is not difficult to prove that if $X^0_{\omega} = X^1_{\omega} = X^1$ for all $\omega \in \Omega$then the existence of coefficients of importance (i.e. $\Lambda^{\Omega} \neq \emptyset$) implies the consistence of Ω in R_q.

Let us consider the principal kinds of importance determined by various classes of functions h.

Lexicographic importance corresponds to a case when Hconsists of all vector-functions h both components of which are functions of "positive shift". In this case the relation $P^{1\succ 2}$ is defined in accordance with (1) and the definition $P^{\succ}$ as follows:

$$x\, P^{1\succ 2}\, y \iff x_1 > y_1,\ x_2 < y_2,$$

and the closure of $R^0 \cup P^{1\succ 2}$ in $\mathcal{R}_q$ brings to a lexicographic (non-strict) order R^L:

$$x\, R^L\, y \iff x_1 > y_1 \ \text{ or } \ (x_1 = y_1,\ x_2 \geq y_2).$$

The only corresponding inequality (2)

$$\lambda_1(x_1 - y_1) > \lambda_2(y_2 - x_2)$$

can not be valid for all x and y in the general case, hence $\Lambda^L = \emptyset$. But if X_1 is discrete and $\inf\{\, x_1 - y_1 \mid x_1, y_1 \in X_1,\ x_1 > y_1 \,\} = \underline{\delta}_1 > 0$ and X_2 is bounded: $\sup\{\, y_2 - x_2 \mid x_2, y_2 \in X_2 \,\} = \overline{\delta}_2 < +\infty$, then Λ^L is formed by all pairs of positive numbers λ_1, λ_2 for which $\lambda_1 \underline{\delta}_1 > \lambda_2 \overline{\delta}_2$. For any $\lambda \in \Lambda^L$ the function $L(x|\lambda)$ represents the relation R^L on X (Lebedev et al., 1971):

$$x\, R^L\, y \iff \lambda_1 x_1 + \lambda_2 x_2 \geq \lambda_1 y_1 + \lambda_2 y_2.$$

Multiplicative importance corresponds to a case when $X \subseteq Re^2_+$, $H = H_\nu = \{ (k_1 t, k_2 t) \mid k_1 = \nu k_2,\ k_1 > 1,\ k_2 > 1 \}$, where the parameter $\nu > 1$. The relation $P^{1 \succ 2}$ is given as follows (here $æ > 1$, $\nu æ > 1$):

$$x\, P^{1 \succ 2}\, y \iff x_1 = \nu æ y_1,\ x_2 = y_2/æ.$$

The inequality (3) looks as $\lambda_1(\nu k_2 - 1)z_1 > \lambda_2(k_2 - 1)z_2$.

If $\inf \{ z_1 \mid z_1 \in X_1 \} = a_1 > 0$, $\sup \{ z_2 \mid z_2 \in X_2 \} = b_2 < +\infty$, and $\nu > 1$ then the set $\Lambda^{1 \succ 2}$ is nonempty: it consists of pairs of positive numbers λ_1, λ_2 complying with the inequality

$$\lambda_1/\lambda_2 > b_2(k_2 - 1)/[a_1(\nu k_2 - 1)].$$

Additive importance corresponds to a case when $H = H_\Delta = \{ h(z) = z + \delta,\ \delta \in \Delta \}$, where the set Δ consists of pairs of positive numbers: $\Delta \subseteq Re^2_+$. In this case (1) and (2) have a form

$$x = (z_1 + \delta_1, z_2), \qquad y = (z_1, z_2 + \delta_2); \tag{5}$$

$$\lambda_1 \delta_1 \geq^\omega \lambda_2 \delta_2, \qquad \omega \in \Omega. \tag{6}$$

It is not difficult to understand that when $\Delta = Re^2_+$ then we have the lexicographic importance. Let us pick out the principal kinds of the additive importance.

Homogeneous importance arises when $X_1 = X_2$, $X^0 \subseteq D^2$, $\Delta = D^2_+$, i.e. when the criteria have a common scale support, X^0 consists of pairs of equal numbers, and Δ is formed by all pairs of equal positive numbers. In accordance with (5) if $\delta_1 = \delta_2$ then the vector estimate y is equal to $\tilde{x}$, where $\tilde{x}$ is turned out from x by transposition of its components. Therefore in accordance with the definitions $I^\sim$ and $P^\succ$ we have

$$x\, I^{1 \sim 2}\, y \qquad \text{iff} \qquad y = \tilde{x} \tag{7}$$

$$x\, P^{1 \succ 2}\, y \qquad \text{iff} \qquad y = \tilde{x},\ \ x_1 > x_2. \tag{8}$$

According to (7) the definition $I^\sim$ corresponds to the axiom of symmetry which is widely known in the theory of decision-making. We have here for (6):

$$\lambda_1 \geq^\omega \lambda_2, \qquad \omega \in \Omega. \tag{9}$$

Proportional importance arises when X_1 and X_2 are intervals and $\Delta = \Delta_\mu = \{ \delta \in D^2_+ \mid \delta_1/\delta_2 = \mu \}$, where the parameter μ is positive. In accordance with (5) the relations $I^{1 \sim 2}$ and $P^{1 \succ 2}$ connect vector estimates such as $(z_1 + \mu\delta_2, z_2)$ and $(z_1, z_2 + \delta_2)$. Therefore if the criteria are of equal importance then μ is a constant rate of substitution. And if one criterion is more important than the other then μ is an upper bound for the rate of substitution. For (6) we have here:

$$\lambda_1 \mu \geq^\omega \lambda_2, \qquad \omega \in \Omega. \tag{10}$$

Graded importance arises when Δ includes a single pair: $\Delta = \{\delta^0\}$. The definition of the relation R^Ω given above is based on the notion of the closure and therefore is non-constructive. If there are judgments of various kinds of the importance then the analysis of consistence of these judgments and construction of the relation R^{Ω} one can fulfill using the general algorithms from (Osipova et al., 1984). But for the importance of any single kind these problems can be resolved much simpler.

Homogeneous importance, $\mathcal{R} = \mathcal{R}_q$.

To begin with let us regard a problem with the criteria of equal importance: $\Omega = \Omega^{\sim} = \{1 \sim 2\}$. In this case (9) has a form of an equality: $\lambda_1 = \lambda_2$, the relation $R_q^{\Omega^{\sim}}$ is given as follows

$$x\, R_q^{\Omega^{\sim}}\, y \iff x_{\downarrow}\, R^0\, y_{\downarrow},$$

where $x_{\downarrow} = (\max\{x_1, x_2\}, \min\{x_1, x_2\})$.

For a problem, in which the first criterion is more important than the second, i.e. $\Omega = \Omega^{\succ} = \{1 \succ 2\}$, (9) turns to be an inequality $\lambda_1 > \lambda_2$, and the relation $R_q^{\Omega^{\succ}}$ is given as follows:

$$x\, R_q^{\Omega^{\succ}}\, y \iff (x\, R_q^{\Omega^{\sim}}\, y \;\text{ if }\; y_1 < y_2;\;\; x\, R^0\, y \;\text{ if }\; y_1 \geq y_2).$$

If in a generalized criterion

$$\Phi^{(s)}(x|\lambda) = (\lambda_1 x_1^s + \lambda_2 x_2^s)^{1/s}, \qquad s \neq 0$$

the importance coefficients λ_1, λ_2 agree with Ω (i.e. $\lambda_1 = \lambda_2$ for $\Omega = \Omega^{\sim}$ and $\lambda_1 > \lambda_2$ for $\Omega = \Omega^{\succ}$), then

$$x\, I_q^{\Omega}\, y \implies \Phi^{(s)}(x|\lambda) = \Phi^{(s)}(y|\lambda); \qquad x\, P_q^{\Omega}\, y \implies \Phi^{(s)}(x|\lambda) > \Phi^{(s)}(y|\lambda).$$

Homogeneous importance, $\mathcal{R} = \mathcal{R}_{qc}$.

The relations $R_{qc}^{\Omega^{\sim}}$ and $R_{qc}^{\Omega^{\succ}}$ are given as follows

$$x\, R_{qc}^{\Omega^{\sim}}\, y \iff \min\{x_1, x_2\} \geq \min\{y_1, y_2\}, \quad x_1 + x_2 \geq y_1 + y_2;$$

$$x\, R_{qc}^{\Omega^{\succ}}\, y \iff (x\, R_{qc}^{\Omega^{\sim}}\, y \;\text{ if }\; y_1 < y_2;\;\; x\, R^0\, y \;\text{ if }\; y_1 \geq y_2).$$

If $s < 1$ then $\Phi^{(s)}(\,\cdot\,|\lambda)$ is strictly quasi-concave, and if λ_1, λ_2 agree with Ω then

$$x\, I_{qc}^{\Omega}\, y \implies \Phi^{(s)}(x|\lambda) = \Phi^{(s)}(y|\lambda); \qquad x\, P_{qc}^{\Omega}\, y \implies \Phi^{(s)}(x|\lambda) > \Phi^{(s)}(y|\lambda).$$

Proportional importance, $X_1 = X_2 = (-\infty, +\infty)$, $\mathcal{R} = \mathcal{R}_q$ $(= \mathcal{R}_{qc})$.

For $\Omega = \Omega^{\succ}(\mu) = \{1 \succ (\mu)2\}$ (9) has a form $\mu\lambda_1 > \lambda_2$ and the set $\Lambda^{\Omega^{\succ}(\mu)}$ turns to be a cone. Also $R_q^{\Omega^{\succ}(\mu)}$ is a cone relations:

$$x\, R^{\Omega^{\succ}(\mu)}\, y \iff x - y \in K^{\Omega^{\succ}(\mu)},$$

where the domination cone $K^{\Omega^{\succ}(\mu)}$ is defined by inequalities $z_2 \geq 0$, $z_1 + \mu z_2 \geq 0$. The cone $\Lambda^{\Omega^{\succ}(\mu)}$ is strictly polar to $K^{\Omega^{\succ}(\mu)}$. For $\Omega = \Omega^{\succ}(\mu', \mu'') = \{1 \succ (\mu')2,\; 2 \succ (\mu'')1\}$ (9) has a form: $\mu'\lambda_1 > \lambda_2$, $\mu''\lambda_2 > \lambda_1$. The preference relation $R_q^{\Omega^{\succ}(\mu',\mu'')}$ is a cone relation:

$$x\, R^{\Omega^{\succ}(\mu',\mu'')}\, y \iff x - y \in K^{\Omega^{\succ}(\mu',\mu'')},$$

where the domination cone is defined by the inequalities: $\mu'' z_1 + z_2 \geq 0$, $z_1 + \mu' z_2 \geq 0$. The information $\Omega^{\succ}(\mu', \mu'')$ is consistent iff $\mu'\mu'' > 1$. In this case $\Lambda^{\Omega^{\succ}(\mu',\mu'')} \neq 0$, and the domination cone $K^{\Omega^{\succ}(\mu',\mu'')}$ is acute.

Graded importance, X_1, X_2 are the sets of integer numbers, δ_1^0 and δ_2^0 are natural numbers, $\mathcal{R} = \mathcal{R}_q$. For $\Omega^\sim = \{1 \sim (\delta^0)2\}$ and $\Omega^\succ = \{1 \succ (\delta^0)2\}$the preference relations are determined as follows

$$x\, R^{\Omega^\sim(\delta^0)}\, y \iff \operatorname{Ent}[(x_1 - y_1)/\delta_1^0] + \operatorname{Ent}[(x_2 - y_2)/\delta_2^0] \geq 0;$$

$$x\, R^{\Omega^\succ(\delta^0)}\, y \iff (x\, R^{\Omega^\sim(\delta^0)}\, y \ \text{ and } \ x_1 \geq y_1).$$

Note that if $\delta_1^0 = 1$ or $\delta_2^0 = 1$, then

$$x\, R^{\Omega^\sim(\delta^0)}\, y \iff \delta_2^0 x_1 + \delta_1^0 x_2 \geq \delta_2^0 y_1 + \delta_1^0 y_2.$$

In (Podinovskii, 1975, 1976, 1978a, 1978b, 1978c, 1979; Menshikova and Podinovskii, 1978; Podinovskii and Polishchuck, 1988) the general case corresponding to $m \geq 2$ is regarded, an ordering of sets of criteria according to their importance is defined and the definition of a comparison of the intensity of exceeding in importance are given. There are the rules of representation of the relation R^Ω (including a representation only by the coefficients of importance). The properties of sets of points non-dominated under P^Ω are also studied.

References

Lebedev, B.D., Podinovskii, V.V. and Stirikovich, R.S. (1971). Optimization Problems with Ordered Criteria. *Economics and Matematical Methods*, No. 4, pp. 612–616, (in Russian).

Menshikova, O.R. and Podinovskii, V.V. (1978). Construction of Preference Relation and Core in Multicriterial Problems with Non-homogeneous Importance-Ordered Criteria. *Zh. Vychisl. Mat. i Mat. Fiz.*, Vol. 28, No. 5, pp. 647–659, (in Russian).

Osipova, V.A., Podinovskii, V.V. and Yashina, N.P. (1984). On Consistent Extension of Preference Relation in Decision-Making Problems. *Zh. Vychisl. Mat. i Mat. Fiz.*, Vol. 24, No. 6, pp. 831–839, (in Russian), (English translation: *USSR Comput. Math. and Math. Phys.*, Vol. 24, No. 3, pp. 128–134).

Podinovskii, V.V. (1975). Multicriterial Problems with Homogeneous Equivalued Criteria. *Zh. Vychisl. Mat. i Mat. Fiz.*, Vol. 15, No. 2, pp. 330–344, (in Russian).

Podinovskii, V.V. (1976). Multicriterial Problem with Importance-Ordered Criteria. *Avtomat. i Telemekh.*, No. 11, pp. 118–127, (in Russian), (English translation: *Automat. Remote Control*, Vol. 37, No. 11, Part 2, pp. 1728–1736).

Podinovskii, V.V. (1978a). Construction of a Set of Effective Strategies in Multicriterial Problems with Criteria that are Ordered with Respect to Importance. *Zh. Vychisl. Mat. i Mat. Fiz.*, Vol. 18, No. 4, pp. 908–915, (in Russian).

Podinovskii, V.V. (1978b). Importance Coefficients of Criteria in Decision-Making Problems. *Avtomat. i Telemekh.*, No. 10, pp. 130–142, (in Russian), (English translation: *Automat. Remote Control*, Vol. 39, No. 11, Part 2, pp. 1514–1524).

Podinovskii, V.V. (1978c). On Relative Importance of Criteria in Multiobjective Decision-Making Problems. In: Multiobjective Decision Problems, Mashinostroeniye, Moscow, pp. 48–92, (in Russian).

Podinovskii, V.V. (1979). An Axiomatic Solution of the Problem of Criteria Importance in Multicriterial Problems. In: Contemporary State of Operation Research Theory, Nauka, Moscow, pp. 117–149, (in Russian).

Podinovskii, V.V. and Polishchuck, L.I. (1988). Convex Multicriterial Problems with Importance-Ordered Criteria. In: The Software of Multicriterial Optimization Problems and Their Application, Yerevan, pp. 166–168, (in Russian).

Generalization of Karlin and Geoffrion Theorems

Leonid I. Polishchuck
Institute of Economics and Industrial Engineering
Siberian Branch of the USSR Academy of Sciences
Novosibirsk, USSR

Abstract

General conditions are obtained for weak and proper efficiency of convex multicriteria problems solutions. These conditions extend well-known Karlin and Geoffrion results for arbitrary concave monotone scalarizing functions. Goal programming problems are considered along with other examples.

Let (X, φ) be convex multicriteria problem formed by convex admissible set $X \subseteq \mathbf{R}^n$ and maximized vector criterion $\varphi : X \to \mathbf{R}^m$ with concave components $\varphi_1, \ldots, \varphi_m$. The following facts known as Karlin and Geoffrion theorems respectively are widely used in MCDM theory and applications (Karlin, 1959; Geoffrion, 1968):

(i) admissible solution $x^* \in X$ is weakly efficient iff there exists coefficients vector $a = (a_1, \ldots, a_m) \in \mathbf{P}_{m-1} = \{ u \in \mathbf{R}_+^m \mid \sum_{j=1}^m u_j = 1 \}$ such that x^* is optimal in the problem

$$\sum_{j=1}^{m} a_j \varphi_j(x) \to \max; \qquad x \in X; \tag{1}$$

(ii) solution $x^* \in X$ is properly efficient iff it is optimal in problem (1) with some positive coefficients vector $a \in \mathbf{P}_{m-1}$.

Characterizing weakly and properly efficient solutions, these theorems give simultaneously necessary and sufficient conditions respectively for the usual efficiency, pointing out the method for construction of Pareto optima by means of maximization of weighted sum of partial criteria. This method, however, has a number of disadvantages from the point of view of computational aspects, possibilities to control efficient solutions choice by coefficients $a_1, \ldots, a_m$ and interpret these coefficients etc. (see, e.g., Polishchuck, 1988). The reasons mentioned prompt to use other variants of general scheme of parametric scalarization which contain the linear weighting method as special case.

The scheme in question stipulates the choice of appropriate scalarizing function $V = V(u_1, \ldots, u_m)$ which generates the family of generalized criteria

$$\Phi_V(x|a) \equiv V(a_1\varphi_1(x), \ldots, a_m\varphi_m(x)) \tag{2}$$

(here $a \in \mathbf{P}_{m-1}$ denotes as before the coefficients vector), and efficient plans are calculated in scalarized problems

$$P_V(a): \qquad V(a_1\varphi_1(x), \ldots, a_m\varphi_m(x)) \to \max; \qquad x \in X. \tag{3}$$

Insertion of parameters vector a into (3) should provide the variety of generalized criteria $\Phi_V(\cdot\,|a)$ adequate to variety of Pareto-optimal solutions of (X, φ) problem. One must be sure, however, that every efficient (weakly, properly) solution of (X, φ) is optimal in problem (3) if the coefficients $a_1, \ldots, a_m$ are properly chosen — if so, the generalized criteria family $\{\, \Phi_V(\cdot\,|a) \mid a \in \mathbf{P}_{m-1} \,\}$ can be called sufficient for (X, φ).

Sufficiency of generalized criteria family of linear weighting method generated according to (2) by linear scalarizing function $V(u) = \sum_{j=1}^{m} u_j$ is guaranteed for convex multicriteria problems by Karlin and Geoffrion theorems. For other scalarizing functions this fact is to be verified each time anew, the verification sometimes appearing laborious enough. Nevertheless the required property holds for the wide class of scalarizing functions distinguished by general conditions of concavity and monotonicity.

Proceeding to exact formulations, one should specify the domain $D \subseteq \mathbf{R}^m$ of function V definition. It is natural to require that the set D should be convex (as convex function domain of definition) and permit the setting of arbitrary coefficient vector $a \in \mathbf{P}_{m-1}$, i.e.

$$(a \otimes u \in D,\ a \in \mathbf{P}_{m-1},\ u \in \mathbf{R}^m) \Rightarrow a' \otimes u \in D, \quad \forall a' \in \mathbf{P}_{m-1}$$

(hereafter $a \otimes u \equiv (a_1u_1, \ldots, a_mu_m)$ denotes the component-wise multiplication of vectors $a, u \in \mathbf{R}^m$). Excluding in addition the degenerate case when D is contained in some coordinate subspace of $\mathbf{R}^m$, one can verify that set D satisfies mentioned requirements iff it is a Cartesian product of m copies of real line or its nonnegative or nonpositive parts:

$$D = I_1 \times \ldots \times I_m, \qquad I_j \in \{\mathbf{R}, \mathbf{R}_+, -\mathbf{R}_+\}, \qquad j = 1, \ldots, m. \tag{4}$$

We formulate now following conditions for weak efficiency, implied by Karlin and Ky Fan (Aubin, 1983) theorems.

Theorem 1. Let (X, φ) be a convex multicriteria problem with $\varphi(X) \subseteq D$, where D is the set of the form (4) and $V : D \to \mathbf{R}$ is convex function continuous on D and monotonically non-decreasing with its variables. It is necessary for the weak efficiency of solution $x^* \in X$, and if V monotonically increases on D-sufficient as well, that the coefficients vector $a^* \in \mathbf{P}_{m-1}$ exists such that x^* is an optimal solution of $P_V(a^*)$ problem.

The proofs of this and the next theorems are omitted here. With $D = \mathbf{R}^m$, $V(u) = \sum_{j=1}^{m} u_j$ one has Karlin theorem as a special case. In general situation theorem 1 states that the sets $\varphi(X)$ and $\{\varphi(x^*)\} + \mathbf{R}_+^m$ can be separated one from another by the function $V(a_1u_1, \ldots, a_mu_m)$ level surface if the coefficients $a_1, \ldots, a_m$ are properly chosen, whatever the concave, continuous and monotone scalarizing function V.

Thus the concavity of scalarizing function plays here the double role: providing the good properties of scalarized problem $P_V(a)$, which can be solved by convex programming methods, it guarantees simultaneously the sufficiency of corresponding generalized criteria family.

The proper efficiency conditions analogous to theorem 1 look as follows.

Theorem 2. Let under theorem 1 conditions function $V : D \to \mathbf{R}$ monotonically increase with its variables. Solution $x^* \in X$ such that $\varphi(x^*) \in \operatorname{int} D$ is properly efficient in (X, φ) problem iff there exists positive coefficients vector $a^* \in \mathbf{P}_{m-1}$ for which x^* is optimal in scalarized problem $P_V(a^*)$.

Geoffrion theorem is the special case of this statement with $D = \mathbf{R}^m$, $V(u) = \sum_{j=1}^m u_j$. The condition $\varphi(x^*) \in \operatorname{int} D$ is satisfied here automatically, but it will be shown later that with $D \neq \mathbf{R}^m$ it can be essential.

Besides the linear scalarizing function $V(u) = \sum_{j=1}^m u_j$, defined on $D = \mathbf{R}^m$, the theorems formulated above can be illustrated by following examples.

Letting

$$V(u) = \min_{j=1,\dots,m} u_j, \qquad D = \mathbf{R}^m \tag{5}$$

we obtain the family of generalized criteria $\Phi_V(x|a) = \min_{j=1,\dots,m} a_j\varphi_j(x)$, which according to theorem 1 permits formulation of necessary (but, generally, not sufficient) conditions for weak efficiency in terms of corresponding problem $P_V(a)$. Here contrary to well-known Germejer theorem (Germejer, 1971) the partial criteria are not supposed to be positive (in the latter case coefficients $a_1, \dots, a_m$ can also be chosen positive, and after that the weak efficiency conditions turn out to be sufficient as well).

Not being strictly monotonical, scalarizing function (5) is unfit for the proper efficient solutions identification. This possibility arises with turning to combined scalarized function

$$V(u) = \sum_{j=1}^m u_j + \alpha \min_{j=1,\dots,m} u_j \qquad (\alpha > 0), \quad D = \mathbf{R}^m \tag{6}$$

(see Wierzbicki, 1981; Gearhart, 1979) which satisfies all conditions of theorems 1,2: for any $\alpha > 0$ and convex multicriteria problem (X, φ), the solution $x^* \in X$ is weakly (properly) efficient iff $x^* \in \operatorname{Argmax}\{\Phi_V(x|a) \mid x \in X\}$ with some $a^* \in \mathbf{P}_{m-1}$ (respectively $a^* \in \mathbf{P}_{m-1},\ a^* > 0$). Analogous criterion for proper efficiency is presented in (Podinovsky and Nogin, 1982).

Scalarizing functions with constant substitution elasticity

$$V(u) = \sum_{j=1}^m u_j^\sigma \qquad (0 < \sigma \leq 1), \quad D = \mathbf{R}_+^m, \tag{7}$$

considered in (Aubin, 1983; Polishchuck, 1982) give necessary and sufficient conditions for weak efficiency (theorem 1), and under additional assumption $\varphi(x^*) \in \operatorname{int} D$, which is equivalent here to positiveness of $\varphi_1(x^*), \dots, \varphi_m(x^*)$ — proper efficiency as well (theorem 2). By scalarizing functions (7) the essence of $\varphi(x^*) \in \operatorname{int} D$ condition can be illustrated: with $\sigma \in (0,1)$, $X = \{x \in \mathbf{R}_+^m \mid \sum_{j=1}^m x_j \leq 1\}$, $\varphi(x) \equiv x$ properly efficient solution $x^* = (1, 0, \dots, 0)$ does not maximize on X function $\sum_{j=1}^m (a_j x_j)^\sigma$ with any $a \in \mathbf{P}_{m-1},\ a > 0$.

Power scalarizing functions

$$V(u) = -\sum_{j=1}^m (-u_j)^s \qquad (s \geq 1), \quad D = -\mathbf{R}_+^m \tag{8}$$

are used for efficient solutions calculation when the vector of non-negative partial criteria is to be minimized (Merkurjev and Moldavsky, 1979). Conditions of theorems 1,2 are also satisfied for these scalarizing functions, which gives one more group of tests for weak and proper efficiency.

Theorems 1,2 are also applicable in goal programming problems. The latter are usually scalarized by certain norm $\|\cdot\|$ in criteria space $\mathbf{R}^m$, satisfying monotonicity conditions

$$u \geqq v \geqq 0 \Rightarrow \|u\| \geq \|v\|. \tag{9}$$

If $u^\nabla \in \mathbf{R}^m$ is a goal vector (formed by "aspirations levels" (Wierzbicki, 1981) for corresponding criteria), where

$$\varphi(x) \leqq u^\nabla, \qquad \forall x \in X, \tag{10}$$

the scalarized problem looks as follows (Gearhart, 1979; Chankong and Haimes, 1983):

$$\left\| a \otimes (u^\nabla - \varphi(x)) \right\| \to \min; \qquad x \in X \tag{11}$$

and consists in choice of solution nearest in criteria space (with respect to $a_1, \ldots, a_m$ coefficients) to the goal vector.

Due to (9), (10) and well-known properties of the norm one can apply theorem 1 after substitution $\tilde{\varphi}(x) \equiv \varphi(x) - u^\nabla$ with $V(u) = -\|u\|$, $D = -\mathbf{R}_+^m$. Thus each weakly efficient solution x^* of convex multicriteria problem (X, φ) is optimal in problem (11) with appropriate $a = a^* \in \mathbf{P}_{m-1}$, whatever the norm $\|\cdot\|$ and goal vector u^∇, satisfying conditions (9), (10) (Polishchuck, 1988).

If, strengthening (9), one assumes strict monotonicity of $\|u\|$ with $u_1, \ldots, u_m$ ($u \in \mathbf{R}_+^m$), then the necessary condition stated for weak efficiency turns out to be sufficient as well. If, further, condition (10) is fulfilled in strengthened form: $\varphi(x) < u^\nabla$, $\forall\, x \in X$, then one obtains the proper efficiency criterion — the latter property is equivalent to optimality of corresponding solution in problem (11) with some $a = a^* \in \mathbf{P}_{m-1}$, $a^* > 0$. In special case of euclidean norm $\|u\| = (\sum_{j=1}^m u_j^2)^{1/2}$ this result was obtained in (Gearhart, 1979).

References

Aubin, J.-P. (1983). Mathematical Methods of Game and Economic Theory. N.Y. et al., North-Holland.

Chankong, V. and Haimes, Y.Y. (1983). Multiobjective Decision Making. Theory and Methodology. N.Y. et al., North-Holland.

Gearhart, W.B. (1979). Compromise solutions and estimation of the noninferior set. *J. Opt. Theory Appl.*, Vol. 28, pp. 29–47.

Geoffrion, A.M. (1968). Proper efficiency and the theory of vector maximization. *J. Math. Anal. Appl.*, Vol. 22, pp. 618–630.

Germejer, Yu.B. (1971). Introduction in Operation Research Theory. Moscow, Nauka, (in Russian).

Karlin, S. (1959). Mathematical Methods and Theory in Games, Programming, and Economics. Addison-Wesley, Reading, Massachusetts.

Merkurjev, V.V. and Moldavsky, M.A. (1979). The family of vector criterion convolutions for finding Pareto set points. *Avtomatika i Telemekhanika*, No. 1, pp. 110–121, (in Russian).

Podinovsky, V.V. and Nogin, V.D. (1982). Pareto-optimal Solutions of Multicriteria Problems. Moscow, Nauka, (in Russian).

Polishchuck, L.I. (1982). On generalized criteria with importance coefficients. *Avtomatika i Telemekhanika*, No. 2, pp. 55–60, (in Russian).

Polishchuck, L.I. (1988). Analysis of Multicriteria Economic Models. Novosibirsk, Nauka, (in Russian).

Wierzbicki, A.P. (1981). A mathematical basis for satisficing decision making. In: Organizations: Multiple Agents with Multiple Criteria. N.Y. et al., Springer, pp. 465–485.

A Realization of Reference Point Method Using the Tchebycheff Distance

I. Popchev, B. Metev, I. Yordanova
Institute of Industrial Cybernetics and Robotics
Bulgarian Academy of Sciences, Sofia, Bulgaria

Abstract

A realization of reference point method for analysis of multiple objective linear programming (MOLP) problems is considered in the paper. The realization uses modified Tchebycheff distance. It is shown that the method obtains Pareto solutions only. The main result of the paper is a theorem that gives a rule for improving the obtained solution with respect to one criterion.

1 Introduction

The investigation of MOLP problems has a almost a 20 years history (Evans, 1984). Different approaches have been used in this period, but the advantages of the interactive procedures for practical applications are unquestionable.

Among other interactive methods, the reference point methods are attractive because they allow the decision maker (DM) to indicate directly points in the criterion space which reflect his preferences. This paper contains some information about a reference point method realization, in which a slight modification of Tchebycheff distance is used. The method yields Pareto points only. The paper shows how the reference point should be changed in order to improve the obtained level of a chosen criterion.

2 Definition of the problem

Let $x \in R^n$ and $x \in S$, where S is the feasible region in the MOLP problem. We will suppose that S is a bounded set. Let us have m nontrivial linear functions $f_i(x)$ defined in S. All coefficients of these functions are nonzero. $f_i(x)$ are considered as partial criteria for optimization and all of them should be maximized. Let $z \in R^m$ be the set of all images of points in S, i.e.:

$$Z = \{\, z \in R^m \mid z_i = f_i(x),\ x \in S,\ i = 1, 2, \ldots, m \,\}$$

The ideal point $z^{\max}$ is defined as follows:

$$z_i^{\max} = \max_{x \in S} f_i(x), \qquad i = 1, 2, \ldots, m$$

The negative ideal point is the point $z^{\min}$, for which:

$$z_i^{\min} = \min_{x \in S} f_i(x), \qquad i = 1, 2, \ldots, m$$

In the following text all used reference points r will satisfy:

$$r_i > z_i^{\max}, \qquad i = 1, 2, \ldots, m$$

The problem under consideration has the following form:

$$\min_x \Big\{ \max_i \{b_i[r_i - f_i(x)]\} - \sum_{j=1}^{m} c_j[f_j(x) - z_j^{\min}] \Big\}$$

$$x \in S, \qquad i = 1, 2, \ldots, m,$$

where:

b_i $(i = 1, 2, \ldots, m)$ are strictly positive real numbers (weights).

$r = (r_1, r_2, \ldots, r_m)$ is the reference point.

c_j $(j = 1, 2, \ldots, m)$ are sufficiently small positive numbers.

The sense of this problem is: given a fixed reference point r^f, to obtain one feasible solution $x^f \in S$, for which the defined minimum is reached. The function, that should be minimized has two main parts. The first one is the ordinary Tchebycheff distance and the second one $\left\{\sum_j c_j[f_j(x) - z_j^{\min}]\right\}$ assures that the obtained solution is nondominated.

The described problem has the following equivalent form as a linear programming problem:

$$\min \ D$$

st

$$\text{(P1)} \qquad D + \sum_{j=1}^{m} c_j E_j(x) + b_i f_i(x) \geq b_i r_i, \qquad i = 1, 2, \ldots, m$$

$$E_j(x) = f_j(x) - z_j^{\min}, \qquad j = 1, 2, \ldots, m$$

$$x \in S$$

This form is used for a reference point method realization in the following way. The user indicates the reference point r^1. The described problem is solved, the corresponding vector x^1 and the values $f_i(x^1)$ $(i = 1, 2, \ldots, m)$ are obtained. Then the DM indicates a new point r^2, obtains the corresponding point x^2 and the vector $[f_1(x^2), f_2(x^2), \ldots, f_m(x^2)]$, etc.

3 Nondominance of the solution

Let as consider the problem $(P1)$ in the following form:

$$\min\ D$$

st

$$\text{(P2)} \qquad D \geq b_i[r_i - f_i(x)] - \sum_{j=1}^{m} c_j E_j(x) = \psi_i(x), \qquad i = 1, 2, \ldots, m$$

$$E_j(x) = f_j(x) - z_j^{\min}, \qquad j = 1, 2, \ldots, m$$

$$x \in S$$

The following notation will be used:

$$u_i(x) = b_i[r_i - f_i(x)]$$

Let $D(x^f)$ is the minimal value that satisfies all constraints of the problem (P2) for a fixed point $x^f \in S$, i.e.:

$$D(x^f) = \max_i \psi_i(x^f)$$

Theorem 1:
Let $x^1 \in S$, $x^2 \in S$, $x^1 \neq x^2$. If $f_i(x^1) \geq f_i(x^2)$ for all i and $f_i(x^1) > f_i(x^2)$ for some i, then $D(x^1) < D(x^2)$.

Proof:
Let:

$$f_i(x^1) > f_i(x^2) \quad \text{for} \quad i \in I_1,$$
$$f_i(x^1) = f_i(x^2) \quad \text{for} \quad i \in I_2,$$
$$I_1 \neq \emptyset, \qquad I_1 \cup I_2 = I = \{1, \ldots, m\}.$$

Then:

$$E_j(x^1) > E_j(x^2) \quad \text{for} \quad j \in I_1,$$
$$E_j(x^1) = E_j(x^2) \quad \text{for} \quad j \in I_2.$$

Therefore:

$$\sum_{j=1}^{m} c_j E_j(x^1) > \sum_{j=1}^{m} c_j E_j(x^2). \tag{1}$$

Further:

$$u_i(x^1) < u_i(x^2) \quad \text{for} \quad i \in I_1, \tag{2}$$
$$u_i(x^1) = u_i(x^2) \quad \text{for} \quad i \in I_2. \tag{3}$$

Relations (1), (2), and (3) assure that:

$$\psi_i(x^1) < \psi_i(x^2) \quad \text{for all} \quad i \in I.$$

Therefore:

$$\max_i \psi_i(x^1) < \max_i \psi_i(x^2).$$

or:

$$D(x^1) < D(x^2).$$

Q.E.D.

This means, that the solution of the problem (P1) (or (P2)) is guaranteed to be a Pareto point.

4 Improving one of the achieved criteria values

The problem (P1) will be examined assuming that $r_i > z_i^{\max}$. Suppose also that the following inequalities are satisfied:

$$b_i[r_i - f_i(x)] - \sum_{j=1}^{m} c_j E_j(x) > 0,$$

$$\text{for all} \quad i \quad \text{and all} \quad x \in S$$

These inequalities should be examined as constraints for c_j. Then, it is possible to choose such numbers δ_1 and δ_2, that the above and the following inequalities hold:

$$0 < \delta_1 \leq \sum_{j=1}^{m} c_j E_j(x) \leq \delta_2, \qquad \text{for all} \qquad x \in S.$$

Suppose that for a fixed reference point r we have obtained the corresponding solution of the problem (P1): the vector x^{ss} and the scalar $D^{\min}$. (We know that the criterion vector $[f_1(x^{ss}), \ldots, f_m(x^{ss})]$ is a nondominated one.) We will accept that the k-th constraint, containing $f_k(x)$ and D, is active in the point x^{ss}, i.e.:

$$D^{\min} = b_k[r_k - f_k(x^{ss})] - \sum_{j=1}^{m} c_j E_j(x^{ss}).$$

Suppose that:

$$f_k(x^{ss}) < \max_{x \in S} f_k(x) = z_k^{\max}.$$

Only in this case it makes sense to solve the following 'pulled' problem:

$$\min\ D^{\Delta}$$

st

$$\text{(P3)} \qquad D^{\Delta} + \sum_{j=1}^{m} c_j E_j(x) + b_i f_i(x) \geq b_i r_i, \qquad i = 1, 2, \ldots, m; \quad i \neq k$$

$$D^{\Delta} + \sum_{j=1}^{m} c_j E_j(x) + b_k f_k(x) \geq b_k(r_k + \varepsilon)$$

$$E_j(x) = f_j(x) - z_j^{\min}, \qquad j = 1, 2, \ldots, m$$

$$\varepsilon > 0$$

$$x \in S$$

The vector $x^{\Delta ss}$ and the scalar $D^{\Delta\min}$ are the solution of (P3). The change of the problem (P1) with (P3) may be interpreted as 'pulling' the reference point with regard to the criterion $f_k(x)$ (for which the corresponding constraint in (P1) in the point x^{ss} is active).

The following assertions can be proved:

$$\text{T2}: \qquad D^{\min} \leq D^{\Delta\min}$$

$$\text{T3}: \qquad D^{\Delta\min} \leq D^{\min} + \varepsilon b_k$$

T2 is almost evident. T3 follows from the equality $D^{\Delta}(x^{ss}) = D^{\min} + \varepsilon b_k$.

Now, the following assumption will be made:

$$\text{S1}: \quad D^{\Delta\min} \leq D^{\min} + \varepsilon b_k - \varepsilon\mu,$$

where μ does not depend on ε and $\mu > 0$.

The following theorem may be proved:

Theorem 4:

Assume S1. Then, ε in (P3) may be chosen in such a way, that the following inequality can be completed:

$$f_k(x^{\Delta ss}) \geq f_k(x^{ss}).$$

Proof:

Because of the activity of the k-th constraint in (P1), the following equality is true:

$$f_k(x^{ss}) = r_k - D^{\min}/b_k - \sum_{j=1}^{m} c_j E_j(x^{ss})/b_k.$$

For the solution of the problem (P3) we have:

$$f_k(x^{\Delta ss}) \geq r_k - D^{\Delta\min}/b_k - \sum_{j=1}^{m} c_j E_j(x^{\Delta ss})/b_k + \varepsilon.$$

Using S1 we get:

$$\begin{aligned} f_k(x^{\Delta ss}) &\geq r_k - (D^{\min} + \varepsilon b_k - \varepsilon\mu)/b_k - \sum_{j=1}^{m} c_j E_j(x^{ss})/b_k + \\ &\quad + \sum_{j=1}^{m} c_j E_j(x^{ss})/b_k - \sum_{j=1}^{m} c_j E_j(x^{\Delta ss})/b_k + \varepsilon = \\ &= f_k(x^{ss}) + w, \end{aligned}$$

where:

$$w = \varepsilon\mu/b_k + \left[\sum_{j=1}^{m} c_j E_j(x^{ss}) - \sum_{j=1}^{m} c_j E_j(x^{\Delta ss})\right] / b_k.$$

It is clear that:

$$w \geq \varepsilon\mu/b_k + (\delta_1 - \delta_2)/b_k.$$

In order to ensure $w \geq 0$ it is sufficient to have:

$$\varepsilon \geq (\delta_2 - \delta_1)/\mu.$$

The same condition guarantees:

$$f_k(x^{\Delta ss}) \geq f_k(x^{ss}).$$

Q.E.D.

5 Comments

It is easy to check that the assumption S1 is true when x is a scalar or a vector with two components. There also exist an indirect way to show that this assumption is true when x is a vector with n components.

In this paper was proved how the solution is changed when the new reference point differs from the first one by one component only corresponding to an active constraint. It is clear that the solution remains unchanged if the increased reference point component corresponds to an inactive constraint and the constraint remains also inactive in the new problem (P3). If the changed reference point component corresponds to an inactive constraint which becomes active in the new problem (P3), then analogous to the Theorem 4 assertion can be proved. In other terms the assumption that the changed reference point component corresponds to an active constraint is not of importance — the obtained solution is not worse with respect to the chosen criterion.

Because of the small parameters c_j, the problems (P1) and (P3) may be interpreted as "perturbed" in comparison with the problem of minimization of the ordinary weighted Tchebycheff distance. Then the theory of small perturbations influence can be used for more detailed investigation of the problems (P1) and (P3).

6 Conclusion

The obtained results illustrate a convenient property of the suggested reference point method realization: improvement in the obtained level of the chosen criterion through a suitable reference point change.

References

Evans, G. (1984). An Overview of Techniques for Solving Multiobjective Mathematical Programs, *Management Science*, Vol. 30, No. 11, pp. 1268–1282.

Lewandowski, A. and Wierzbicki, A. (1987). Theory, Software and Testing Examples for DSS, IIASA, WM-87-26.

Nakayama, H. (1985). On the Components in Interactive Multiobjective Programming Methods. In M. Grauer, M. Thompson, A. Wierzbicki, eds., Plural Rationality and Interactive Decision Analysis, Springer-Verlag.

Steuer, R. (1986). Multiple Criteria Optimization: Theory, Computation and Application, John Wiley & Sons, N.Y.

A New Algorithm for Vector Optimization Problem

B.N. Pshenichnyj, A.A. Sosnovskij
V.M. Glushkov Institute of Cybernetics
Ukrainian Academy of Sciences
Kiev, USSR

A vector optimization problem arises at the first step of the general procedure of decision making when from the whole set of admissible alternatives some set is singled out each element of which meets the requirements of an efficient solution.

Let the functions $f_i(x)$, $i = 1, 2, \ldots, m$, $x \in \mathbf{R}^n$ be specified. They form the vector criterion $f(x) = (f_1(x), \ldots, f_m(x))^T$ of some multi-objective optimization problem

$$\begin{gathered} \min\ f(x) \qquad \text{under constraints} \\ x \in \mathbf{X}_g = \{\, x \in \mathbf{R}^n \mid g_j(x) \leq 0,\ j = 1, 2, \ldots, l \,\}. \end{gathered} \tag{1.1}$$

When considering only linear functions $f_i(x) = c_i^T x$, $i = 1, 2, \ldots, m$ and $g_j(x) = a_j^T x$, $j = 1, 2, \ldots, l$ we may already say about a linear vector optimization problem

$$\begin{gathered} \min\ Cx \qquad \text{under constraints} \\ x \in \mathbf{X} = \{\, x \in \mathbf{R}^n \mid Ax \leq b \,\}. \end{gathered} \tag{1.2}$$

where C is the $(n \times m)$-matrix of objective function coefficients, A is the $(n \times l)$-matrix of constraint coefficients and b is the l-vector.

The majority of the known strategies of vector optimization problem solution consists in characterizing the effective solutions in terms of optimal solutions of some corresponding scalar optimization problem. The difference in methods of reducing the vector criterion $f(x)$ to one objective function defines the conventional classification of approaches to solution of the vector optimization problem (Mikhalevich and Volkovich, 1982).

This paper investigates the properties of methods based on the idea of linearization in the aspect of multi-objective optimization. These properties are characterized in terms of weak efficient, efficient (Pareto-optimal) and proper efficient solutions. The obtained results hold true for a linear case.

Let us introduce some definitions (Podinovskij and Nogin, 1982). The solutions x^* is called weak efficient (efficient) if there is no such x that $x \neq x^*$, $f(x) < (\leq)\ f(x^*)$ is satisfied.

Admissible solution x^* is called proper efficient if it is efficient and there exists such positive number M that for any $i = 1, 2, \ldots, m$ and $x \in \mathbf{X}_g$ for which the inequality $f_i(x^*) > f_i(x)$ is satisfied and some $\nu \in \{1, \ldots, m\}$ such that $f_\nu(x^*) < f_\nu(x)$ the inequality $[(f_i(x^*) - f_i(x))/(f_\nu(x) - f_\nu(x^*))] \leq M$ is satisfied.

1 Basic assumptions

Let us consider some modification of the linearization algorithm (Pshenichnyj, 1983) based on necessary conditions of the efficiency of the problem (1.1). Relate the auxiliary problem to every point $x \in \mathbf{X}_g$:

$$\begin{aligned} &\min_{p,\xi} \left\{\xi + \tfrac{1}{2}\|p\|^2\right\} \qquad \text{under constraints} \\ &\nabla f_i(x)p \leq \xi, \qquad i \in I, \\ &\nabla g_j(x)p + g_j(x) \leq 0, \qquad j \in J, \end{aligned} \tag{1.3}$$

where $I = \{1, 2, \ldots, m\}$, $J = \{1, 2, \ldots, l\}$.

Introduce the following assumptions. Let us exist such $N > 0$ that

a) for some $i \in I$ the set $\Omega_N = \{x \in \mathbf{R}^n \mid f_i(x) + NG(x) \leq C_i\}$ where $C_i = f_i(x_0) + NG(x_0)$, $G(x) = \max\{0, g_1(x), \ldots, g_l(x)\}$ is limited;

b) gradients $\nabla f_i(x)$, $i \in I$ and $\nabla g_j(x)$, $j \in J$ in Ω_N satisfy Lipschitz condition with the constant L;

c) there exist such Lagrange multipliers of the problem (1.3) $v_j(x)$, $j \in J$ that $\sum_{j\in J} v_j(x) \leq N$ and the latter is solvable with respect to $p \in \mathbf{R}^n$ for any $x \in \Omega_N$.

It is easy to see that the problem (1.3) is equivalent to the following convex programming problem (Pshenichnyj, 1983):

$$\begin{aligned} &\min_{p} \left\{\tfrac{1}{2}\|p\|^2 + \max_{i\in I}\{\nabla f_i(x)p\}\right\} \qquad \text{under constraints} \\ &\nabla g_j(x)p + g_j(x) \leq 0, \qquad j \in J. \end{aligned}$$

Now we write necessary and sufficient conditions relating the minimum point to Lagrange multipliers of the problem (1.3). Note that the Lagrange function has the form:

$$\begin{aligned} L(p, \xi, u, v) &= \xi + \frac{1}{2}\|p\|^2 + \sum_{i\in I} u_i(x)[\nabla f_i(x)p - \xi] + \\ &\quad + \sum_{j\in J} v_j(x)[\nabla g_j(x)p + g_j(x)] = \\ &= \xi\Big[1 - \sum_{i\in I} u_i(x)\Big] + \frac{1}{2}\|p\| + \sum_{i\in I} u_i(x)\nabla f_i(x)p + \\ &\quad + \sum_{j\in J} v_j(x)[g_j(x) + \nabla g_j(x)p]. \end{aligned}$$

There exist such $u_i(x) \geq 0,\ i \in I,\ v_j(x) \geq 0,\ j \in J$ that

$$u_i(x)[\nabla f_i(x)p - \xi] = 0, \qquad i \in I, \tag{1.4}$$

$$v_j(x)[\nabla g_j(x)p + g_j(x)] = 0, \qquad j \in J, \tag{1.5}$$

$$\sum_{i \in I} u_i(x) = 1, \tag{1.6}$$

$$p(x) + \sum_{i \in I} u_i(x)\nabla f_i(x) + \sum_{j \in J} v_j(x)\nabla g_j(x) = 0. \tag{1.7}$$

2 Algorithm formulations and principal results

Now we formulate the computational procedure for solving the vector optimization problem. Let x^0 be the initial approximation and $\varepsilon,\ 0 < \varepsilon < 1$ be chosen. Let the point x^k be already obtained. Then

1. We solve the auxiliary problem (1.3) for $x = x^*$ and find $p^k = p(x^k)$.

2. We find the first value of $s = 1, 2, \ldots$ for which the inequality

$$\max_{i \in I} [f_i(x + \alpha p) - f_i(x)] + NG(x + \alpha p) \leq NG(x) - \alpha\varepsilon\|p\|^2. \tag{1.8}$$

   will be satisfied for $\alpha = (1/2)^s$. If such s is found then assume $\alpha_k = 2^{-s}$, $x^{k+1} = x^k + \alpha_k p^k$.

From the assumption about the continuity of vector function $f(x)$ components on the non-empty compact Ω_N the existence of all kinds of effective points follows (Podinovskij and Nogin, 1982). Now we formulate the first convergence theorem.

Theorem 1. Let the assumptions a)–c) of section 1 be satisfied. In addition, let the Cottle regularity condition be satisfied at any limiting point x^* genarated by the proposed algorithm: there is such point $p \in \mathbf{R}^n$ that for any $j \in J(x^*) = \{\, j \in J \mid g_j(x^*) = 0 \,\}$ the inequality $\nabla g_j(x^*)p < 0$ is satisfied. Then the point x^* satisfies necessary conditions of weak efficiency and $\|p^k\| \to 0$.

Because of the limited size of this paper the proofs of the given statement are not presented here. The interested reader can find them in (Pshenichnyj and Sosnovskij, 1987). Note only that it is based on reducing the extremum necessary conditions of the problem (1.3) in solution to the equation

$$\sum_{i \in I} u_i^*(x^*)\nabla f_i(x^*) + \sum_{j \in J(x^*)} v_j^*(x^*)\nabla g_j(x^*) = 0. \tag{1.9}$$

The latter together with the Cottle regularity condition and the relation (1.6) correspond to the necessary condition of weak efficiency of the point x^* formulated by Da Cunha-Polak-Geoffrion theorem presented in (Podinovskij and Nogin, 1982). It is shown there that under some assumptions about the convex functions $f(x)$ and $g(x)$ the expression

(1.9) corresponding to sufficient optimal conditions. Therefore the following statements hold true (Pshenichnyj and Sosnovskij, 1987).

Corollary 1.1. Let the vector-function $f(x)$ be pseudoconvex and functions $g_j(x)$ for any $j \in J(x^*)$ be quasi-convex. Then necessary and sufficient conditions of weak efficiency are satisfied at the point x^* in the conditions of Theorem 1.

Corollary 1.2. If, in addition, the strict quasi-convexity of the vector-function $f(x)$ and the convexity of $\mathbf{X}_g$ are assumed then the point x^* may be stated to satisfy necessary and sufficient conditions of efficiency (Pareto-optimum).

According to the second part of the Da Cunha-Polak-Geoffrion theorem the expression (1.9) is a necessary condition of the proper efficiency of solution x^* if the condition $u_i^* > 0$ for any $i \in I$ and $\sum_{i \in I} u_i = 1$ is satisfied in it. It is not difficult to see that in general case the suggested algorithm does not provide Lagrange positive multipliers. However the latter may be guaranteed by having imposed some conditions of generalized regularity (CGR). Namely, let $\nabla f_i(x)$, $i \neq \nu$, $i \in I$ and $\nabla g_j(x)$, $j \in J(x^*)$ be linearly independent at the point $x^* \in \mathbf{X}_g$ for any $\nu \in I$. It is clear that the given condition is a certain generalization of the usual condition of regularity of a single-objective mathematical programming problem in form of the condition of a linear independence of the gradients of active constraints. The assumption made makes it possible to strengthen the result of Theorem 1.

Theorem 2. Let the major assumptions a)–c) of section 1 be fulfilled. And let the CGR be satisfied at the limiting point x^* of the above algorithm. Then the point x^* satisfies the necessary conditions of proper efficiency (Pshenichnyj and Sosnovskij, 1987).

The work (Podinovskij and Nogin, 1982) shows that if Lagrange multipliers u_i^*, $i \in I$ are positive and the equality (1.9) is fulfilled then the assumption about the pseudo-convex function $f(x)$ is already insufficient for proper efficiency of solution as it is the case in Corollary 1.2.

Corollary 2.1. Let the conditions of the previous Theorem be satisfied. Then if the vector-function $f(x)$ is convex and functions $g_j(x)$ are quasi-convex for any $j \in J(x^*)$ then the point x^* satisfies the necessary and sufficient conditions of proper efficiency (Pshenichnyj and Sosnovskij, 1987).

So far we have considered the problem of vector optimization in the general form, i.e., when the objective and constraint functions have a nonlinear character. It is in this case that we should expect the greatest effect from application of the suggested algorithm. However its use in the linear case is quite reasonable (Pshenichnyj, 1983). The linear problem of vector optimization possesses its specific peculiarities that allow the conditions of algorithm application be substantially simplified and therefore the finite results be strengthened.

First of all we note that not all assumptions a)–c) of section 1 remain necessary. Thus, the condition b) is satisfied automatically. The convexity of objective and constraints functions allows to speak both of necessary and sufficient conditions of efficiency of the limiting point x^* of the algorithm.

If we take into account the fact that in the linear case the sets of Pareto-optimal and proper efficient solutions coincide (Podinovskij and Nogin, 1982) then the given characteristic of the point x^* is complete. Here one can take from the above-mentioned two conditions a weaker one.

Theorem 3. Let assumption a) and c) in Section 1 be fulfilled. In addition, let the Cottle regularity condition be satisfied at any limiting point x^* genarated by the suggested algorithm: there is such point $p \in \mathbf{R}^n$ that for any $j \in J(x^*) = \{j \in J \mid a_j^T x^* = b\}$ the inequality $a_j^T p < 0$ is fulfilled. Then the point x^* satisfies necessary and sufficient conditions of efficiency. The algorithm itself converges in the finite number of step.

The first part of this theorem is a simple corollary of Theorem 1 proved for vector optimization problem in the general form. The proof of the algorithm finiteness is in principle based on the ideas of the proof of the finiteness of the general method of linearization for linear programming problem and shown in (Sosnovskij, 1988).

References

Mikhalevich, V.S. and Volkovich, V.L. (1982). Computational methods of study and design of complex systems. Moscow, Nauka, 288p.

Podinovskij, V.V. and Nogin, V.D. (1982). Pareto-optimal solutions of multi-objective problems. Moscow, Nauka, 256p.

Pshenichnyj, B.N. (1983). Linearization method. Moscow, Nauka, 136p.

Pshenichnyj, B.N. and Sosnovskij, A.A. (1987). Linearization method for solving multi-objective optimization problem. *Kibernetika*, No. 6, pp. 107–109.

Sosnovskij, A.A. (1988). On one algorithm of solving linear problem of vector optimization. Methods and Software Tools of Optimization, Simulation and Construction of Systems. Sb. nauchn. tr., V.M. Glushkov Institute of Cybernetics, Acad. of Sciences of the Ukr. SSR, Kiev, pp. 42–47.

Multicriteria Optimization in Synthesis Problems of Distributed Control Systems

V. F. Shostak
Kharkov Institute of Radiolectronics
Kharkov, USSR

The distributed control systems (DCS) are the joint of micro-computers (microprocessing stations, personal computers) combined into local computing networks (LCN). The results of information processing in these networks are realised at the stations under control by means of communication devices with object and displayed at operators terminates. The distribution of control tasks among data processing DCS nodes, territorial distribution of terminal devices and computing units are features of such systems. Advanced reliability and productivity, low cost of data communication lines and other known dignities of DCS allow to distinguish them as the most perspective trend of computer-aided control systems development (Schenbrot et al., 1985).

The task of DCS synthesis includes the points of selection of type modules structure from base complex of software-hardware means, definition of ties structure among them, originating from pacularity of controlled processes, demands laying claims to systems and given criteria of efficiency, such as, fast response, reliability and DCS (Shostak, 1987). Thus, synthesing of optimal DCS is coupled with the task of multicriteria optimisation.

The solution of this task is carried out on the basis of formal DCS description in form of functional graph $G_Z(Q_Z, S_Z)$ and graph of regional distribution of system (topological graph) $G_P(Q_P, S_P)$. Functional graph is defined by plurality of Q_Z^* nodes corresponding to control tasks $\{Z_q\}$ and by plurality of nodes Q_Z^{**} that correspond to external sources and users of information (terminal devices). For instance, sensors, displays, equipment at controlled objects $Q_Z = Q_Z^* \cup Q_Z^{**}$, $Q_Z^* \cap Q_Z^{**} = \emptyset$. The arcs of functional graph S_Z describe the data flows among tasks and also tasks and terminal devices.

It is advisable show the plurality of topological graph Q_P nodes in form of unintersecting sets combination $Q_P = Q_P^{**} \cup Q_P^*$, where Q_P^{**} – nodes corresponding to points of terminal devices location, Q_P^* corresponds to probable location of computing complexes. The representation $Q_Z^{**} \rightarrow Q_P^{**}$ is obvious bijection. The ribs S_P correspond to rational data communication lines route among Q_P nodes; thus, for each of rib there are some defined factors, such as length of route.

Every control task Z_q is represented by set of initial data X_q and solution results Y_q, by estimate of evarage rate of task solution W_q, maximum admittable time δ_q of service request to task Z_q solution and by vector of factors C_q that characterise labour consuming nature of task solution level of demands to reliability and certainty of solution.

$$
\begin{aligned}
&Z_q = \{X_q, Y_q, W_q, \delta_q, C_q\},\\
&X_q \in \{X_{iq}\} \quad i \in \theta_q, \quad Y_q = \{Y_{qj}\} \quad j \in \theta_q^*, \quad X_{iq} = Y_{iq},
\end{aligned} \tag{1}
$$

where

X_{iq} – initial data of task Z_q originating from i-th node of G_Z graph,

Y_{qj} – the results of task Z_q solution originating into j-th node of graph G_Z,

θ_q, θ_q^* – sets of order indexes of graph G_Z nodes, connected with q-th node by entering and issuing arcs, respectively.

The problem of DCS synthesing is in defining of representation:

$$
f_{\alpha 0} : Q_Z^* \to Q_P \tag{2}
$$

and corresponding to it complex of software-hardware means optimal by given criteria $\bar{\varphi} = \{\varphi_k\}$, $k \in I$, which characterise, for instance, cost, fast response and reliability of DCS.

In terms of the theory of multicriteria optimisation (Michalevich, Volkovich, 1982) this task may be written:

$$
\min_{\alpha \in A} \sum_{k \in I} \rho_k \cdot W_k(\varphi_k(\alpha)), \tag{3}
$$

$$
\sum_{k \in I} \rho_k = 1, \quad \forall\, k \in I,\ \rho_k > 0, \tag{4}
$$

where

$\bar{\rho} = \{\rho_k\}$ – vector of criteria preference $\bar{\varphi}$,

A – set of possible representations (2),

$W_K(\varphi_k(\alpha))$ – function of criteria reduction to dimension less form.

For the case of φ_k optimisation

$$
W_K(\varphi_k(\alpha)) = \frac{\varphi_k^0 - \varphi_k(\alpha)}{\varphi_k^0 - \varphi_{k\,\min}}, \tag{5}
$$

where

φ_k^0, $\varphi_{k\,\min}$ – respectively the most and the least meanings φ_k in the set of alternative variants $\alpha \in A$.

Every alternative variant $\alpha_\nu \in A$ of representation (2) is characterised by the family of unintersecting subsets $\{W_{(\nu)j}\}$, $j = \overline{1, N_\nu}$ of task Z_q. Along with it $Z_q \in W_{(\nu)j}$ is solved by hardware means in j-th node of graph G_P. So far as in general case not in all nodes Q_P it is advisable to have means for solving tasks, $N_\nu \leq |Q_P|$.

Thus, the optimal variant of representation (2) describes the best concerning criteria distribution of tasks $\{Z_q\}$ among nodes of graph G_P, which in its turn is the basis for optimal of composition and structure of DCS software-hardware means originating from attributes (1) of every task Z_q.

The advantage of such approach is that DCS synthesis is realised only on the basis of given aims and tasks of object controlling, demands, conditions and peculiarities of controlled processes with minimum usage of heuristic assumptions.

A great number of alternative variants A and absence of analytical dependencies $W_k(\varphi_k(\alpha))$ makes it difficult to solve the tasks (3), (4) using known methods. In connection with this for solving the task it is proposed to use the principle of local data processing (Schenbrot, et al., 1985 and Witte, 1978), on the basis of which it is advisable to solve any task Z_q on the node of topological graph G_P at which appear (us a result of solving other tasks introduced from terminal devices) initial data of this task $X_q = \{X_{iq}\}$.

The data blocks $\{X_{iq}\}$ may originate from different task and terminal devices. Thus, on the principle of local data processing is based the advisability of solving task Z_q at that node of graph G_P from which originate the most initial data flow

$$\lambda_{iq} = W_q \cdot |X_{iq}| \tag{6}$$

On that way the principle of local data processing gives the conclusion that the family of subsets $\{W^{(1)}_{(\nu)j}\}$ of tasks Z_q, corresponding to optimal representation (2), satisfy the condition of joining in a subset $W^{(1)}_{(\nu)j}$ of task with the largest meaning of relation estimate

$$b^{(1)}_{iq} = \lambda_{iq} + \lambda_{qi} \tag{7}$$

Therefore, in order to single out the family of tasks $\{W^{(1)}_{(\nu)j}\}$ subsets it is advisable to use the method of extreme grouping (Braverman, 1970) on the basis of relation estimate (7). The extreme grouping of relation estimate tasks (7) allows to minimise the data flows, transferred by channels of LCN DCS and thus to reduce the cost of data transfer means, to raise fast response of system and also the reliability of it at the expense of reduction of failures during data transfer.

The reliability of DCS is defined not only by channels of data transfer, thus under tasks grouping it is advisable also to single out the family of subsets $\{W^{(2)}_{(\nu)j}\}$ on the basis demands to reliability and certainty of obtained results. Along with it for solving of subsets of the most primary tasks it is necessary to foresee especially reliable and money cost means at the expense of cost reduction of solving other tasks.

In the capacity of relation estimate $b^{(2)}_{iq}$ when singling out the subset of tasks $\{W^{(2)}_{(\nu)j}\}$ on criterion of reliability it is possible to use, in particular, the linear functions of the type:

$$b^{(2)}_{iq} = R - V \cdot |h_q - h_r| \tag{8}$$

where

h_q, h_r – estimates of tasks Z_q and Z_r reliability level,

R, V – constant values.

The families of subsets of tasks $\{W^{(1)}_{(\nu)j}\}$ and $\{W^{(2)}_{(\nu)j}\}$ singled out under grouping on the basis of relation estimates (7) and (8), respectively, not coincide in general case, so there appears the need in correction them.

The technique tasks (3), (4) solving on the basis of offered approach, taking into account the criterion φ_1 of minimum data exchange among the tasks subsets and the criterion of reliability maximisation φ_2 under limiting of system cost, include the following steps:

1. Definition, according to formula (7), the relation estimates $b^{(1)}_{iq}$ among nodes Q_2 of functional graph G_Z.

2. Singling out by method of extreme grouping of the family of subsets $\{W^{(2)}_{(\nu)j}\}$, $j = \overline{1, N^{(2)}_\nu}$ that corresponds to criterion φ_1.

3. Estimation criterion φ^0_1 meaning for the family of task subsets, singled out in point 2.

4. Definition, according to formula (8) the relation estimates $b^{(2)}_{iq}$ among nodes Q_Z.

5. Singling out, by method of extreme grouping, the family of subsets $\{W^{(2)}_{(\nu)j}\}$, $j = \overline{1, N^{(2)}_\nu}$ which corresponds to criterion φ_2.

6. The estimation of meaning φ^0_2 for the family of task subsets, singled out at point 5.

7. Construction of tasks subsets

$$\begin{aligned} \{\Theta^{(1,2)}_{Pl}\}, \qquad & P = \overline{1, N^{(1)}}, \quad l = \overline{1, N^{(2)}_\nu}, \\ \Theta^{(1,2)}_{Pl} \quad = \quad & W^{(1)}_{(\nu)P} \cap W^{(2)}_{(\nu)l} \end{aligned} \tag{9}$$

8. Definition of relation estimate $\eta^{(1,2)}$ among subsets of tasks (9) taking into account two criteria φ_1 and φ_2 with coefficients $\beta_1 > 0$, $\beta_2 > 0$, $\beta_1 + \beta_2 = 1$

$$\eta^{(1,2)}_{Pl.sn} = \sum_{\substack{Z_q \in \Theta_{Pl} \\ Z_r \in \Theta_{sn}}} \left(\beta_1 \cdot b^{(1)}_{qr} + \beta_2 \cdot b^{(2)}_{qr}\right) \tag{10}$$

9. Singling out, by method of extreme grouping of tasks (9) subsets, the family of subsets of tasks $\alpha^{(1,2)}_\nu = \{W^{(1,2)}_{(\nu)j}\}$, $j = \overline{1, N^{(1,2)}_\nu}$, $N^{(1,2)}_\nu \le |Q_P|$.

10. Estimation of meaning of $\sum_{k \in I} \rho_k \cdot W_k(\varphi_k(\alpha^{(1,2)}_\nu))$ and analysis of received solution. If necessary the correction $\overline{\beta} = \{\beta_1, \beta_2\}$ is carried out taking into account the values $\{\rho_k \cdot W_k(\varphi_k(\alpha^{(1,2)}_\nu))\}$ in (3) and transition to point 8.

It should be noted that the method of extremal grouping in general case guarantees only approximate task solution of singling out the family of subsets $\{W_{(\nu)j}^{(h)}\}$ that is optimal under criterion φ_h. However, insufficient computing costs of described methodic allow to consider a number of different variants of task subsets formulation with the aim of improving received solutions.

The advantage of presented approach is that DCS synthesis is carried out by formal method with regard for a number of contradictory criteria originating from description of pacularities of controlled object in form of topological graph G_P, demands to the system, aims and tasks of control, described by graph G_Z and attributes of tasks (1).

In connection with it the approach introduces the methodological base of creating instrumental programming means for computer-aided DCS design.

Presented methodic was utilized when designing computer integrated manufacturing (CIM).

References

Braverman, A. M. (1970). Methods of extremal parameters grouping and task of sufficient factors singling out. *Automation and telemechanics.* Nauka, 1, pp. 123–139.

Michalevich, V. S., Volkovich, V. L. (1982). Computing methods on research and design of complex systems. Nauka, 286 p.

Schenbrot, I. M., Antropov, M. W., Davidenko, K. J. (1985). Distributed computer-aided control systems for production engineering. Under A. A. Levin's editorship. Energoatomizdat, 240 p.

Shostak, V. F. (1987). System approach in synthesis tasks of distributed control systems. Analysis and synthesis of distributed information controlled: Thesis of reports and communication of republic seminar. Tbilisi, Metsniereba, pp. 104–105.

Witte, J. (1978). Lokale Speicher in Datenverarbeitungssystemen. Elektronische Rechenanlagen, Bd20, 3, pp. 109–114.

Interactive Multiobjective Optimization Based on Ordinal Regression

Roman Słowiński
Institute of Computing Science
Technical University of Poznan
60-965 Poznan, Poland

1 Introduction

We introduce a new approach to solving optimization problems formulated in terms of multiobjective mathematical programming. It relies on the concepts of ordinal regression and fuzzy outranking relation in a finite set of efficient points. The presented methodology applies both to linear and to nonlinear (including discrete) programming problems. Using the common terminology (cf. Steuer, 1986), the new approach could be classified as the contracting cone method with a visual constructive interaction.

A usual problem in multiobjective mathematical programming is how to focus on a subset of efficient points of greatest interest to the DM, in early stages of the interactive process. Existing interactive procedures represent either descriptive or constructive approach (Bouyssou, 1984). The first one postulates that DM's preferences are stable and that he can control them in a logical and coherent way. The interactive process consists then in description and exhibition of a pre-existing global utility function. Thus, rather than speaking about interaction, one should speak here about iterative acquisition of information. In the latter approach, the DM's preferences are not supposed to pre-exist but they can evolve in the interactive process. The system of DM's preferences is constructed basing on some regularities of preferential attitudes, accepting however their instability and incompleteness. The interactive process is a learning process in the trial and error spirit.

We advocate the constructive interactive approach to multiobjective mathematical programming which take into account the inherent imprecision, uncertainty and inaccurate determination.

Because of the different kinds of uncertainty, the DM may not be able to compare solutions, especially in early stages of the interactive procedure, using **indifference** and **strict preference** relations only. For this reason, it is useful to distinguish 3 following relations for any pair a, b of alternatives:

- $a\ P\ b$, i.e. a is **significantly preferred** to b,
- $a\ I\ b$, i.e. a and b are **equivalent**,
- $a\ Q\ b$, i.e. a is **weakly preferred** to b.

Introduction of Q corresponds to the definition of indifference and preference thresholds q_g and p_g, respectively, for criterion g. If g is to be maximized,

- $a\ I\ b \Leftrightarrow -q_g(g(a)) \leq g(a) - g(b) \leq q_g(g(b))$,
- $a\ Q\ b \Leftrightarrow q_g(g(b)) < g(a) - g(b) \leq p_g(g(b))$,
- $a\ P\ b \Leftrightarrow p_g(g(b)) < g(a) - g(b)$.

The following logical conditions are imposed on q_g and p_g, supposing a minimum of coherence in DM's preferences:

$$\frac{q_g(g(b)) - q_g(g(a))}{g(b) - g(a)} \geq -1, \qquad \frac{p_g(g(b)) - p_g(g(a))}{g(b) - g(a)} \geq -1.$$

Criterion g involving indifference and preference thresholds is called **pseudo-criterion.**

Roy (1985) proposes moreover to handle situations where the DM is not able or don't want to make distinction between $a\ P\ b$, $a\ Q\ b$ and $a\ I\ b$. He uses a grouped relation S called **outranking relation.** $a\ S\ b$ means that a is at least as good as b, *non* $a\ S\ b$ and *non* $b\ S\ a$ means that a and b are incomparable.

The outranking relation can be valued between 0 and 1 to express the strength of affirmation "a outranks b". It is then called **fuzzy outranking relation.**

We shall use the fuzzy outranking relation to model DM's preferences in early stages of the interactive procedure for solving multiobjective programming problems. In the course of this procedure, the relation is translated into an efficient region of greatest interest to the DM, using an ordinal regression method. This region is then scanned by the DM and if he selects the best compromise point the procedure stops, otherwise further reduction of the efficient region is performed which best fits the DM's preferences.

2 General scheme of the interactive procedure

The general multiobjective programming problem is formulated as

$$\max\{\, g_1(x) = z_1\}$$

$$\cdots\cdots$$

$$\max\{\, g_k(x) = z_k\}$$

$$\text{s.t. } x \in D$$

where the g_i need not be linear and D need not be convex. It is assumed that each objective is bounded over D and that there does not exist a point in D at which all objectives are simultaneously maximized.

The general scheme of the proposed interactive procedure is the following.

Step 1. Generation of a small (finite) subset A of efficient points (from 10 to 50), as representative as possible of the efficient set E.

Step 2. Construction of a fuzzy outranking relation S in subset A.

Step 3. Construction of two complete preorders $\overline{P}$, $\underline{P}$ in subset A using so-called descending and ascending distillations of S.

Step 4. Assessment of two scalarizing functions (augmented weighted Chebyshev metrics) compatible with $\overline{P}$ and $\underline{P}$, respectively, using an ordinal regression method.

Step 5. Interactive exploration of an efficient subset $\tilde{E} \subset E$ defined by diagonal directions of the scalarizing functions assessed in Step 4, then STOP, if a best-compromise point has been found, or generation of a new subset $A \subset \tilde{E}$ and return to Step 2, otherwise.

3 Description of steps

3.1 Step 1

The generation of sample A of (weakly) efficient points can be performed using one of several existing methods, for example the method by Choo and Atkins (1980) which we applied already in a different context (cf. Jacquet-Lagrèze et al., 1987, and Słowiński, 1986). The techniques by Morse (1980), Tõrn (1980) and Steuer (1986, ch.14) can also be used in this step. The last one is the most general and can compute weakly efficient points of integer and nonlinear multiobjective programming problems. It is recommended to equalize the ranges of the coordinates of efficient points prior to the generation.

3.2 Steps 2 and 3

Sample A is then presented to the DM who is taking part in construction of a fuzzy outranking relation S in set A. The construction of S proceeds as in ELECTRE III (Roy, 1978).

When comparing the elements of A, the DM considers the objectives as pseudo-criteria. Fuzzy outranking relation S is characterized by the definition of an outranking degree associating each pair of alternatives $a, b \in A$ with a number $0 \leq d(a,b) \leq 1$; d is a measure of credibility of the outranking "$a\ S\ b$".

For each pseudo-criterion g_j, two indices have to be calculated first for any pair $a, b \in A$: concordance index $c_j(a,b)$ and discordance index $D_j(a,b)$. The former expresses to what extent the evaluation of a and b on g_j is concordant with assertion "a is at least as good as b". The latter indicates the strength of its opposition against this assertion. $D_j(a,b)$ involves a veto threshold v_j , i.e. the bound beyond which the opposition to the hypothesis "$a\ S\ b$" is sufficiently motivated. The definition of both indices is given graphically in Fig. 1.

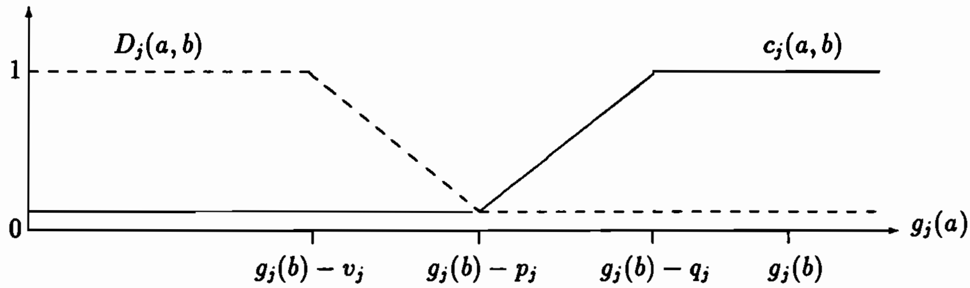


Fig. 1. Concordance and discordance indices for $a, b \in A$ and criterion g_j

The partial concordance indices are then aggregated taking into account relative importance of criteria defined be indices k_j,

$$C(a,b) = \sum_{j=1}^{k} k_j c_j(a,b) / \sum_{j=1}^{k} k_j$$

The degree of credibility $d(a,b)$ is obtained from the global concordance index, weakened by discordance indices (up to the point of its annulment),

$$d(a,b) = C(a,b) \prod_{j \in J} \frac{1 - D_j(a,b)}{1 - C(a,b)}, \qquad J = \{ j : \; D_j(a,b) > C(a,b) \}$$

The goal of the next step is to derive two complete preorders in A, as different as possible, from the fuzzy outranking relation. Preorder $\overline{P}$ is obtained in a descending way, i.e. selecting first the best alternatives, then the following, until the worse. Preorder $\underline{P}$ is obtained in an ascending way, i.e. the selection process starts with the worse alternatives and ends with the best ones. In ELECTRE III, this procedure is called distillation. $\overline{P}$ and $\underline{P}$ are different in general — this difference reflects the range of DM's hesitancy in the present stage of problem solving.

3.3 Step 4

Given two complete preorders $\overline{P}$ and $\underline{P}$ in set A, two scalarizing functions are assessed, as compatible as possible with $\overline{P}$ and $\underline{P}$, respectively. As the scalarizing function we use the augmented weighted Chebyshev metric

$$s(z^*, z, \underline{\lambda}, \rho) = \max_{j=\overline{1,k}} \{\lambda_j (z_j^* - z_j)\} + \rho \sum_{j=1}^{k} (z_j^* - z_j) \tag{1}$$

where $z_j^* = \max\{g_j(x) : \; x \in D\} + \varepsilon_j$, $\varepsilon_j \geq 0$ is moderately small, $\lambda_j \geq 0$, $\sum_{j=1}^{k} \lambda_j = 1$, and ρ is a sufficiently small positive number. We have chosen this metric because of two main advantages: it can be used to find (weakly) efficient points in a nonconvex set and its assessment according to ordinal regression reduces to linear programming.

The idea of ordinal regression was already used to assess a piecewise-linear utility function for multiobjective linear programming (cf. Jacquet-Lagrèze et al., 1987).

Let $\hat{s}_{\overline{P}}$ denote the scalarizing function perfectly compatible with $\overline{P}$, $s_{\overline{P}}$ — the scalarizing function being assessed and σ — an approximation error. For every $a \in A$,

$$\hat{s}_{\overline{P}}(z^*, g(a), \underline{\lambda}, \rho) = s_{\overline{P}}(z^*, g(a), \underline{\lambda}, \rho) + \sigma(a)$$

Assuming a small threshold $\delta > 0$, one can express all relations which constitute $\overline{P}$

$$a\ P\ b \;\Leftrightarrow\; s_{\overline{P}}(z^*, g(b), \underline{\lambda}, \rho) - s_{\overline{P}}(z^*, g(a), \underline{\lambda}, \rho) + \sigma(b) - \sigma(a) \geq \delta \tag{2}$$

$$a\ I\ b \;\Leftrightarrow\; -\delta < s_{\overline{P}}(z^*, g(b), \lambda, \rho) - s_{\overline{P}}(z^*, g(a), \lambda, \rho) + \sigma(b) - \sigma(a) < \delta \tag{3}$$

Substituting (1) to right-hand side of (2) and (3), we get

$$y_b - y_a + \rho \sum_{j=1}^{k} (z_j^* - g_j(b)) - \rho \sum_{j=1}^{k} (z_j^* - g_j(a)) + \sigma(b) - \sigma(a) \geq \delta \tag{4}$$

$$-\delta < y_b - y_a + \rho \sum_{j=1}^{k} (z_j^* - g_j(b)) - \rho \sum_{j=1}^{k} (z_j^* - g_j(a)) + \sigma(b) - \sigma(a) < \delta \tag{5}$$

$$y_b \geq \lambda_j (z_j^* - g_j(b)), \qquad j = 1, \ldots, k \tag{6}$$

$$y_a \geq \lambda_j (z_j^* - g_j(a)), \qquad j = 1, \ldots, k \tag{7}$$

Parameters λ and ρ of the scalarizing function $s_{\overline{P}}$ follow from the linear program

$$\begin{array}{ll} \min & \left\{ \sum\limits_{a \in A} \sigma(a) \right\} \\ \text{s.t.} & \left. \begin{array}{l} (4) \text{ if } a\ P\ b \\ (5) \text{ if } a\ I\ b \\ (6), (7) \end{array} \right\} \begin{array}{l} \text{for all pairs } a, b \in A \text{ such that} \\ a \text{ and } b \text{ are "consecutive" in } \overline{P} \end{array} \\ & \sigma(a) \geq 0 \qquad \text{for all } a \in A \\ & 0 \leq \rho \leq 0.001, \qquad \lambda_j \geq 0, \qquad j = 1, \ldots, k, \qquad \sum\limits_{j=1}^{k} \lambda_j = 1 \end{array} \tag{8}$$

LP problem (8) has $2|A| + k + 1$ variables and at most $(k+2)|A|$ constraints, so it is better to solve its dual form. Analogical LP problem can be set up for $s_{\underline{P}}$. An illustration of the assessment step is shown in Fig. 2.

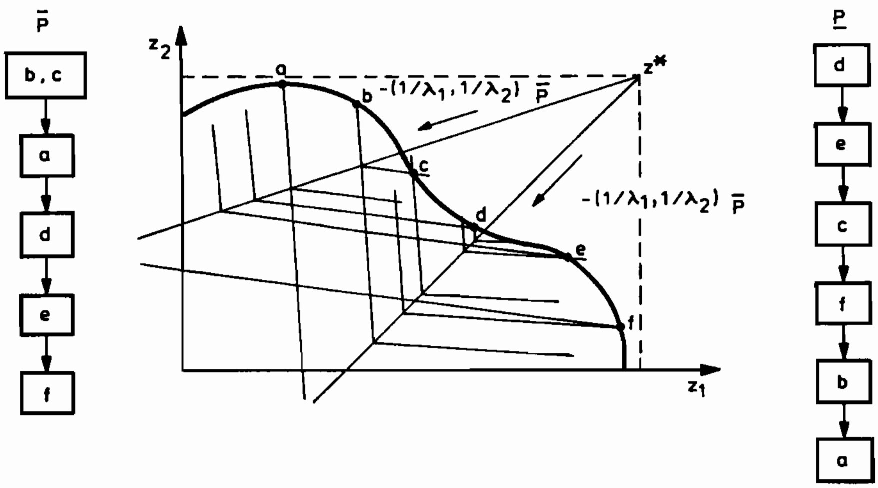


Fig. 2. Scalarizing functions perfectly compatible with $\overline{P}$ and $\underline{P}$

3.4 Step 5

Diagonal directions of Chebyshev metrics $s_{\overline{P}}$ and $s_{\underline{P}}$,

$$-(1/\lambda_1, \ldots, 1/\lambda_2)_{\overline{P}} \quad \text{and} \quad -(1/\lambda_1, \ldots, 1/\lambda_2)_{\underline{P}}$$

are then used to define generators of a cone with a vertex in z^*. They can be obtained from rotation of diagonal directions round the axis $-[2/(\underline{\lambda}_{\overline{P}} + \underline{\lambda}_{\underline{P}})]$. Another simple procedure is composed of following steps

Step 5.1. Compute efficient points z^A and z^B minimizing $s_{\overline{P}}$ and $s_{\underline{P}}$, respectively.

Step 5.2. Solve the augmented weighted Chebyshev program

$$\min \{ s(z^*, z, (\underline{\lambda}_{\overline{P}}, \underline{\lambda}_{\underline{P}})/2, \rho) \}, \quad \text{s.t.} \quad s(z) \geq s(z^A), \quad s(z) \geq s(z^B)$$

The obtained point is denoted by $\tilde{z}$.

Step 5.3. Solve k single-objective programs ($j = 1, \ldots, k$)

$$\min \{ g_j(x) \}, \quad \text{s.t.} \quad g_i(x) \geq \tilde{z}_i, \quad i = 1, \ldots, k; \quad i \neq j, \quad x \in D$$

One obtains k (weakly) efficient points $z^1, z^2, \ldots, z^k$.

Step 5.4. Calculate k generators of the cone with a vertex in z^*

$$\begin{array}{rcl} -(1/\lambda_1^1, 1/\lambda_2^1, \ldots, 1/\lambda_k^1) & = & (z_1^1 - z_1^*, z_2^1 - z_2^*, \ldots, z_k^1 - z_k^*) \\ \cdots\cdots\cdots & \cdots & \cdots\cdots\cdots \\ -(1/\lambda_1^k, 1/\lambda_2^k, \ldots, 1/\lambda_k^k) & = & (z_1^k - z_1^*, z_2^k - z_2^*, \ldots, z_k^k - z_k^*) \end{array}$$

then STOP.

The cone generates an efficient subset $\tilde{E} \subset E$ corresponding to current preferences of the DM, modelled by S. The scanning of subset $\tilde{E}$ consists in calculating and showing to the DM trajectories of objectives between the points $z^A - z^C$, $z^B - z^C$, $z^1 - z^C$, and so on, until $z^k - z^C$, where z^C is an efficient point following from the augmented weighted Chebyshev program

$$\min \{ s(z^*, z, (\underline{\lambda}_{\overline{P}}, \underline{\lambda}_{\underline{P}})/2, \rho) \}, \qquad \text{s.t. } z = g(x), \quad x \in D$$

Other scanning methods can also be used (cf. Korhonen and Laakso, 1986, Lewandowski and Wierzbicki, 1988, Steuer, 1986, ch.14). If the DM finds $\tilde{E}$ still too large, he may restart the construction of S for some $A \subset \tilde{E}$, possibly with finer thresholds q_j and p_j.

4 Final remarks

The focusing of a cone defining a most interesting efficient region corresponds to the reduction of hesitancy in the evaluation of efficient points. The principle of the presented methodology relies on the observation that the focusing procedure should take into account different kinds of uncertainty (here modelled by an outranking relation) which is successively reduced due to the learning of the DM in the interactive process of solving the multiobjective program.

References

Bouyssou, D. (1984). Approches descriptives et constructives d'aide à la décision: fondaments et comparaison. Thèse de 3e cycle, Université de Paris-Dauphine.

Choo, E. U., Atkins, D. R. (1980). An interactive algorithm for multicriteria programming. *Computers & Operations Res.* 7, pp. 81–87.

Jacquet-Lagrèze, E., Meziani, R., Słowiński, R. (1987). MOLP with an interactive assessment of a piecewise-linear utility function. *Europ. J. Operational Res.* 31, pp. 350–357.

Korhonen, P., Laakso, J. (1986). A visual interactive method for solving the multiple criteria problem. *Europ. J. Operational. Res.* 24, pp. 277–287.

Lewandowski, A., Wierzbicki, A. (1988). Aspiration based decision analysis and support — Part I. WP-88-03, International Institute for Applied Systems Analysis, Laxenburg, Austria.

Morse, J. N. (1980). Reducing the size of the nondominated set: pruning and clustering. *Computers & Operations Res.* 7, pp. 55–66.

Roy, B. (1978). ELECTRE III — un algorithme de classement fondé sur une représentation floue des préférences en présence de critères multiples. Cahiers du CERO 20, pp. 3–24.

Roy, B. (1985). Méthodologie Multicritère d'Aide à la Décision. Economica, Paris.

Słowiński, R. (1986). A multicriteria fuzzy linear programming method for water supply system development planning. *Fuzzy Sets & Systems* 19, p.. 213–237.

Steuer, R. (1986). Multiple Criteria Optimization — Theory, Computation and Application. J. Wiley, New York.

Törn, A. (1980). A sampling-search-clustering approach for exploring the feasible/efficient solutions of MCDM problems. *Computers & Operations Res.* 7, pp. 67–80.

Problems with Multiple Objectives and Ill-Posed Problems

I.M. Sobol
Institute of Mathematics and Mechanics
Moscow, USSR

Abstract

Stabilizers of ill-posed problems may be considered as new independent criteria. Conversely, ill-posed problems may originate from multicriterial ones.

Regularization of ill-posed problems

Consider a computational problem: to find the solution of a generally non-linear equation

$$A(z) = f; \qquad z \in G. \tag{1}$$

For simplicity we suppose that $G \subset \mathbf{R}^n$ and $f \in \mathbf{R}^m$ but cases $n = \infty$ or/and $m = \infty$ are not excluded.

Problem (1) is often reduced to the following problem:

$$\rho(A(z), f) \to \min; \qquad z \in G, \tag{2}$$

where ρ is a metric in the f space. But in many cases it turns out that the last problem is unstable or in other words ill-posed. A well-known regularization method (Tikhonov and Arsenin, 1977) is to choose a stabilizing term $\Omega(z)$ and to calculate the minimum $z = z_\alpha$ of the sum

$$\rho(A(z), f) + \alpha\Omega(z) \to \min; \qquad z \in G. \tag{3}$$

Here $\alpha > 0$ is a small parameter that somehow must be fixed: if α is too large the error $z_\alpha - z$ will be large; if α is too small the calculation will be unstable. The method is rather widely used though the problem of selecting α is not completely solved.

Multicriterial interpretation

In (Sobol, 1986) two objective functions

$$\phi_1 \equiv \rho(A(z), f), \qquad \phi_2 \equiv \Omega(z)$$

are introduced and a two-criterial problem is formulated:

$$\phi_1(z) \to \min, \qquad \phi_2(z) \to \min; \qquad z \in G. \tag{4}$$

Let E be the set of efficient (non-dominated) points of the problem (4). The following theorem is proved in (Sobol, 1986).

Theorem. If $z = z_\alpha$ is the solution of (3) corresponding to an arbitrary $\alpha > 0$ then $z_\alpha \in E$.

Curiously enough in the theory of ill-posed problems we encounter the problem $\phi_1 + \alpha\phi_2 \to \min$ as well as problems connected with conditional minima of ϕ_1 or ϕ_2 (Rakhmatulina, 1972). But the two-criterial problem (4) has never been formulated until 1985. And geometrical illustrations in the two-dimensional criteria space (ϕ_1, ϕ_2) are regarded as something new.

The criteria space

The functions $\phi_1(z)$ and $\phi_2(z)$ define a mapping

$$z \to (\phi_1, \phi_2) \tag{5}$$

of the set or feasible points G into a set $\tilde{G}$ in the criteria space. Under the transformation (5) the set E of efficient points is transformed into a set $\tilde{E}$ that is usually called the Pareto set (Fig. 1). Form Fig. 1 the geometrical interpretation of (3) is quite obvious: point B is the image of z_α. Fig. 2 shows that in general the set of all z_α is not equal to E: the images of all z_α fill the arcs B_*B_1 and B_2B_3.

We turn now to one of the most popular strategies for selecting α (Morozov, 1966). Here the main assumption is that the computational error in $\|f\|$ is $\sim \delta$ whereas the error in $A(z)$ is much smaller. Then the error in ϕ_1 is $\sim \delta$ too. Since values $\phi_1 < \delta$ are not reliable it seems reasonable to select α form the equation $\phi_1 = \delta$ or explicitly

$$\rho(A(z_\alpha), f) = \delta. \tag{6}$$

A traditional approach is to investigate analytically conditions of existence and unicity of a solution $\alpha = \alpha(\delta)$ of (6).

Now the geometrical interpretation of (6) is clear from Fig. 2. And the problem can be decomposed into several partial problems.

1. Given δ, one has to find a point B from $\tilde{E}$ with an abscissa δ. The unicity of B is implied by the strict monotonicity of $\tilde{E}$. But Fig. 3 indicate that such a point may not exist (if $\delta \in \Delta$).

2. After the point B has been found one has to define α. Fig. 2 shows that such a value may not exist (if $B \in B_1B_2$).

3. At last one has to find a point z_α satisfying the relation

$$(\phi_1(z_\alpha), \phi_2(z_\alpha)) = B.$$

   The unicity of z_α can be guaranteed only if the mapping $E \to \tilde{E}$ is a one-to-one correspondence: $E \leftrightarrow \tilde{E}$.

Although (5) is a mapping from $\mathbf{R}^n$ into $\mathbf{R}^2$, the situation $E \leftrightarrow \tilde{E}$ is not an exception. This may be concluded from the following theorem (Sobol, 1986).

Theorem. Suppose that $\phi_1(z)$ and $\phi_2(z)$ are continuous functions defined in G and hypersurfaces

$$\phi_1(z) = c_1, \qquad \phi_2(z) = c_2 \tag{7}$$

have no common singular points. Assume that pairs of hypersurfaces (7) may be non-intersecting, or intersecting, or have only one common points. Let $z \in E$, $z' \in E$ and $z \neq z'$. If at least one of these points belongs to $\operatorname{int} G$ then

$$\phi_1(z') \neq \phi_1(z), \qquad \phi_2(z') \neq \phi_2(z).$$

Thus, different efficient points z and z' have different images in $\tilde{E}$. While formulating the conditions of the theorem we have excluded the situation where hypersurfaces (7) do not intersect but have more than one common point.

Suggestions

From the preceding considerations it seems that it may be interesting to investigate:

1. The structure of $\tilde{E}$ in case $\phi_1(z)$ is multimodal but $\phi_2(z)$ is rather simple (e.g. $\phi_2 = \|z - z_0\|$ where z_0 is a given point).

2. Various sufficient conditions for $E \leftrightarrow \tilde{E}$.

There are problems (Tikhonov and Arsenin, 1977) where it seems reasonable to consid several stabilizers $\Omega_1(z), \ldots, \Omega_k(z)$ simultaneously. In that case our interpretation will be multicriterial rather than two-criterial. And it is not necessary to combine these stabilizers beforehand.

How ill-posed problems originate

We have seen that the regularization of ill-posed problems may be interpreted as the introduction of several objective functions. Now a kind of an inverse statement: ill-posed problems sometimes arise when we try to replace multicriterial problems by one-criterial.

Example 1. Consider an algebraic system

$$\left.\begin{aligned} 10(x_1^2 - x_2) &= 0 \\ 1 - x_1 &= 0 \end{aligned}\right\}$$

that has one solution $x_1 = 1,\ x_2 = 1$. This solution can be easily calculated by numerical methods. But if we replace the system by a problem of type (2)

$$100(x_1^2 - x_2)^2 + (1 - x_1)^2 \to \min$$

we obtain on the left the well-known "banana-shaped" function (Rosenbrock and Storey, 1966) that is often used as an example of a function that behaves badly in the vicinity of its minimum.

Example 2. In (Tikhonov and Arsenin, 1977) (chapter 8, §1) a problem of optimal planning is discussed, where one has to choose a plan to keep a plant running at maximum capacity. The problem is unstable: for comparatively close values of the load ($\pm 1\%$) the number of different items to be produced fluctuates from zero to several hundreds.

It is clear that here one doesn't have to invent new stabilizing terms. One has to restore those additional objective functions that were arbitrarily omitted. Undoubtedly such additional criteria exist: you cannot completely ignore what items are produced.

Example 3. The reduction of (1) to (2) depends on the choice of the metric ρ in the f space. However in many problems (e.g. identification, data processing, etc.) various components f_j of f represent very different physical quantities with different orders of magnitude and different errors. The usual advice is to introduce weights:

$$\rho(f, f') = \left\{ \sum_{j=1}^{m} \lambda_j (f_j - f_j')^2 \right\}^{1/2}.$$

But in fact nobody knows which are the right λ_j. An arbitrary choice of weights often leads to bad results.

In that case a more adequate approach is to consider a problem with multiple objectives

$$\Psi_j \equiv |A(z)_j - f_j| \to \min, \qquad 1 \le j \le m \tag{8}$$

(Of course equations corresponding to similar quantities can be combined to reduce the number of objectives). Analysing the values of $\Psi_1, \ldots, \Psi_m$ we obtained much more information than from the values of one eclectic function $\phi_1 = \rho(A(z), f)$. This is especially important when there is no z_* satisfying $\rho(A(z_*), f) = 0$ and the least values of ϕ_1 depend strongly on the choice of weights. It can be easily verified that a solution z of (2) corresponding to a fixed set of positive weights λ_j is an efficient solution of (8).

Mathematicians readly acknowledge the adequacy of (8). But how to tackle such problems?

Sometimes a possibly way is to use the Parameter Spase Investigation. An identification problem of type (8) has been solved by that method in (Matusov et al., 1985).

On the parameter space investigation (LP$_\tau$-method)

The main purpose of that interactive method (Sobol and Statnikov, 1981, 1985) is to select criterial constraint values that define a non-empty set of admissible points $D \subset G$

(a solution is called admissible if it is acceptable with respect to all the criteria). The same method enables us to approximate the Pareto set, to estimate the linear dependence of objective functions, etc. Whenever possible such an investigation should precede any optimization attempts. (I cannot help citing (Spronk, 1985): "...multiple criteria decision methods are much less tools of optimization than they are tools of learning and communication...")

More than once users expressed their surprise at the excellent performance of that crude method. But only recently a persuasive explanation has been given: it is quality of the trial points computed from LP_τ-sequences (Sobol, 1969). The advantage of these sequences has been suggested by their very uniform distribution. However recent investigations (Sobol, 1982, 1988) show that a systematic search with the aid of these points is the best way of dealing with functions $f(z)$ that satisfy Lipschitz conditions

$$|f(z) - f(z')| \leq \sum_{j=1}^{m} L_j |z_j - z'_j|$$

with arbitrary non-negative unknown constants L_j.

Subroutines to generate LP_τ-sequences can be found in (Sobol and Statnikov, 1981; Sobol, 1979; Bratley and Fox, 1988). A program package CRIT has been written (Gorbunov-Posadov et al., 1985) that supports the method of Parameter Space Investigation in problems of optimum design.

References

Bratley, P. and Fox, B.L. (1988). Implementing Sobol's quasirandom sequence generator. *ACM Math. Software*, Vol. 14, No. 1, pp. 88–100.

Gorbunov-Posadov, M.M., Ermakov, A.V., Karpov, V.Ya., Matyokin M.P., Sobol, I.M. (1985). CRIT — applied program package solving multicriterial problems in optimum design of machines. Preprint, Inst. of Appl. Math., USSR Acad. of Sciences, Moscow. No. 175, (in Russian).

Matusov, I.B., Perminov, M.D., Rymalova, S.I., Sobol, I.M., Statnikov, R.B. (1985). Verification of a mathematical model of an object — the main problem in optimum design. In: Problems and methods of decision making in organizational management systems. Moscow, Inst. for Systems Studies, pp. 13–23, (in Russian).

Morozov, V.A. (1966). On the regularization of ill-posed problems and the selection of regularization parameter. *USSR Comput. Math. and Math. Phys.*, Vol. 6, No. 1, pp. 170–175.

Rakhmatulina, A.Kh. (1972). Ill-posed problems and methods of solving. Preprint, Inst. of Anst. of Appl. Math., USSR Acad. of Sciences, Moscow, No. 1, (in Russian).

Rosenbrock, H.H. and Storey, C. (1966). Computational technique for chemical engineers. Oxford, Pergamon Press.

Sobol, I.M. (1969). Multidimensional quadrature formulas and Haar functions. Moscow, Nauka, (in Russian).

Sobol, I.M. (1979). On the systematic search in a hypercube. *SIAM J. Numer. Anal.*, Vol. 16, No. 5, pp. 790–793, (correction in No. 6).

Sobol, I.M. (1982). On an estimate of the accuracy of a simple multidimensional search. *Soviet Math. Dokl.*, Vol. 266, No. 3, pp. 569–572.

Sobol, I.M. (1986). Multicriterial interpretation of a regularization method for ill-posed problems. *USSR Comput. Math. and Math. Phys.*, Vol. 26, No. 6, pp. 805–812.

Sobol, I.M. (1988). On the search for extremal values of a multi-variable function satisfying a general Lipschitz condition. *USSR Comput. Math. and Math. Phys.*, Vol. 28. No. 4, pp. 483–491.

Sobol, I.M. and Statnikov, R.B. (1981). Selection of optimal parameters in problems with several criteria. Moscow, Nauka, (in Russian).

Sobol, I.M. and Statnikov, R.B. (1985). Multicriteria problems in machine design. In: Multicriterial mathematical programming problems. Moscow, Inst. for Systems Studies, pp. 39–45.

Spronk, J. (1985). Multiple objective decision methods in macro-economic planning. In: Multicriterial mathematical programming problems. Moscow, Inst. for Systems, pp. 45–51.

Tikhonov, A.N. and Arsenin, V.Ya. (1977). Solutions of ill-posed problems. N.Y., Winston & sons.

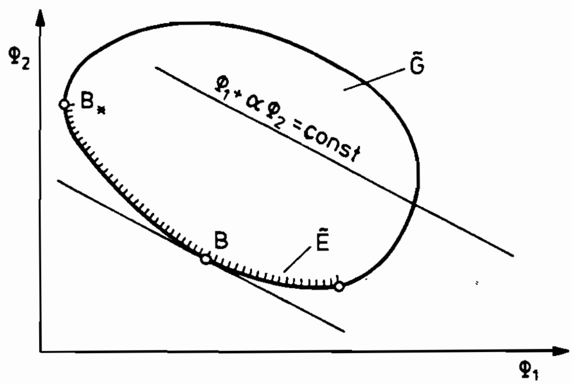


Figure 1

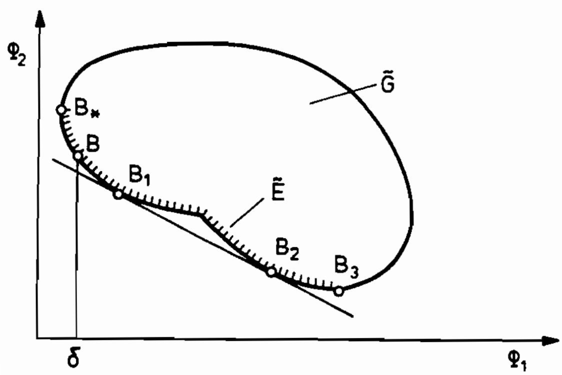


Figure 2

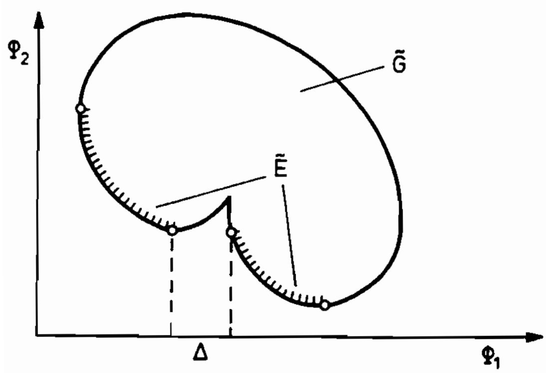


Figure 3

DSS – ES Integration Problems

Ivan Stanchev
Laboratory for Systems Analysis and Management
Karl Marx University
Sofia, Bulgaria

The fast development of the application of decision support systems (DSS) during the last few years as well as the ever-increasing requirements of the users towards them brought about new concepts of increasing the intelligence of these systems (Chen Ye-Sho, 1988). Simultaneously, the debates on the real capacities of the expert systems for decision support (Martins Gary, 1984) made the specialists in this field seek the collaboration of the DSS designers. That collaboration is based on the mutual benefit of the two sides from the creation and setting up of qualitatively new system products of the integration between DSS and ES.

Decision support systems are systems that use computers and help the manager in decision-making when solving ill-structured tasks. They provide support without substituting the decision-maker and raise the efficiency of decision-making. The major objective of DSS is to sustain the activities of decision-making with the help of a computer and to provide for the combination of objective and subjective elements (the personal opinion of the decision-maker and the result of the processing of the material).

The major objective of the expert systems is to provide the means, i.e. new methods in the field of artificial intelligence, for retrieval representation and use of experts' knowledge. They are applied in decision-making as well, and that is why the meaning of these two terms overlaps.

ES specialists, and especially ES designers, discussed in detail the issue whether DSS and ES are really very close to one another. As a matter of fact there are many differences that turn the DSS into a not too big "Horse of Troy" with regard to ES.

Basically, in decision support systems the emphasis is laid on such decisions that involve easily computerized and mathematical structures, as well as statistical patterns, but in all cases the manager's opinion is the decisive factor. The real benefits of implementing DSS are in broadening the scope od decision-making, its possibilities with the general aim of raising labour efficiency and perfecting the manager's work. A very efficient help is to put under the manager's control the means of sustaining the process which does not need preliminary aim definition and the automation of the decision-making process. Using a DSS, the manager imperceptibly broadens his interests and increases his knowledge through using the computer. This is exactly the opposite to what happens when using MIS (management information system) which lays the emphasis on information that has already been found, analyzed and interpreted and thus offered for decision-making. Decision support systems provide a broader-based approach to decision-making than the one offered by management information systems.

The main function of an expert system is consultation. An expert's opinion is taken and then it is shaped and encoded in a computer. The user, who is usually not an expert, can now consult the computer and not the expert in person. Expert systems can be used in industry as well in providing engineers with adequate help in structuring complex management issues. So, the first attempts were at setting up complex simulation models (simulating different situations) by engineers that are not experts.

There are differences in using the two systems. For instance, expert systems are linked to research activities and its users are generally senior researchers or researchers in a special, narrow field. Expert systems have been created to reflect and get ready for utilization the knowledge and experience of very famous experts.

DSS, on the other hand, is applied mainly in organizations and industrial enterprises. Its users are managers of the middle and high level. The DSS user can very often supply the decision by himself and sometimes he had taken part in designing and putting the DSS together. DSS is designed for a specific group of users. The DSS user can have an ingenious approach to a certain problem but the dialogue with a DSS would give him a wider choice of solutions, data and model structures that he can use in the further investigation of the problem (William, 1984). ES users have less opportunity to voice an initiative than DSS users. The usage of suitable components from the knowledge base is directly under the control of the retrieval mechanism, it is based on its controlling mechanisms, that is based on problem euristics and depends on the knowledge base that it contains. ES is used for consulting while DSS users can also control the functioning of different components.

Another difference between DSS and ES is the type of programming language used in the two systems. The language used in DSS is usually the conventional high-level language, for example FORTRAN, BASIC, etc. or a unconventional modelling language system that was created for DSS in the first place. That is proved by the very characteristics of the models — typical mathematical algorithms (Gevarter, 1983).

ES usually involves artificial intelligence languages, as for example LISP, PROLOG, etc. that are more efficient in representing and processing symbolic information that is necessary in putting together and applying expert systems.

Regardless of the common approach, the methodology of setting up both systems is also very different. DSS and ES design is in itself an interactive process of setting up and developing prototypes. The main bulk of the information and the general requirements of operation are set up during the development of a working prototype. The demonstration of the prototype proves the necessity to perfect and broaden the scope of the prototype, and to put altogether a new upgrade of the whole system. This process is iterated till reaching the required level of efficiency. This approach to DSS design gives a chance of adapting the system perfectly to its users' needs. It also ensures the flexibility of the system and the possibility to put it into use before it is completed.

In ES design the prototype approach gives the opportunity to perfect and broaden the knowledge base. This is a crucial decision for the researcher — to define the stage of completeness of the system and the quality of the facts and euristic methods in the knowledge base. Typical means for solving the problem are prototypes that have knowledge bases that concern simple, iterative processes till the knowledge base is broadened enough so that it can be tested through real, practical problem. This process is more time-consuming in ES design than in DSS design.

Examining integration problems bearing in mind the above-mentioned differences, it becomes obvious that some ideas for ES can be used in DSS design, too, including retrieval possibilities, euristic structures and decision-making models (Fordyce, 1985). That is exactly the main topic of discussion in this paper.

The research and the accumulated experience up to the present show that not all decision support systems develop in one and the same way (Sol, 1985). Regardless of this, it is very important to the DSS concept with the aim to define the trends of its present and future development and to raise the efficiency of the process of decision-making.

Some researchers try to project DSS on the principles of cognitive psychology. For example, in (Keen, 1983) there is a concept that the process of summing up these principles is still at its initial stage. Besides, in (Keen, 1983) it is reaffirmed that the critical factor in the decision-making process is the one that ensures the presence of euristics in problem-solving.

Another aspect of this issue is the link between the research in artificial intelligence and DSS design. In (Cats-Baril and Huber, 1984) it is stated that the huge possibilities of preserving and restructuring information cannot guarantee its organizational and social usefulness. The development of intelligent systems on data bases do not solve this problem. Just the opposite, it can even make the present situation worse if low-quality information is put into the system.

No matter that DSS can provide the connection between the convention information-processing methods and knowledge construction, one should never forget that Expert Systems are nothing but a summary of accumulated experience.

In summing up already accumulated experience and knowledge one should view the creative process in a completely new way.

Figure 1 shows the new DSS design scheme that is different from the schemes of Sprangue, Bonczek and Sol. This system displays the concept of DSS generation, or, to put it in a different way, the environment of DSS design. A part of the process of solving ill-structured problems is the choice of suitable paradigms for the construction and the modelling method that will be the fundamentals for defining the concept of the problem and for the actual solving of the problem. This choice would later determine the methodology and theoretical framework of the problem — solving. The idea is to put together the given paradigm of the construction and the DSS generation concepts in existence at the moment. The combination of these two can be shaped as a reference system. This reference system in itself is a structured multitude of means that can be used as prompts or as modelling environment in the process of problem-solving.

The reference system would be used as a "conceptualization environment" at the initial stage, too. Using the language system and its knowledge base, it formulates the basic elements used in designing the system. Besides the reference system, a concept-defining problem model is offered as well as an empirical model of decision-searching and the system of implementation of this model.

The using of a reference system has the following assets:

- using the construction paradigms in concept-building one can follow the stages of

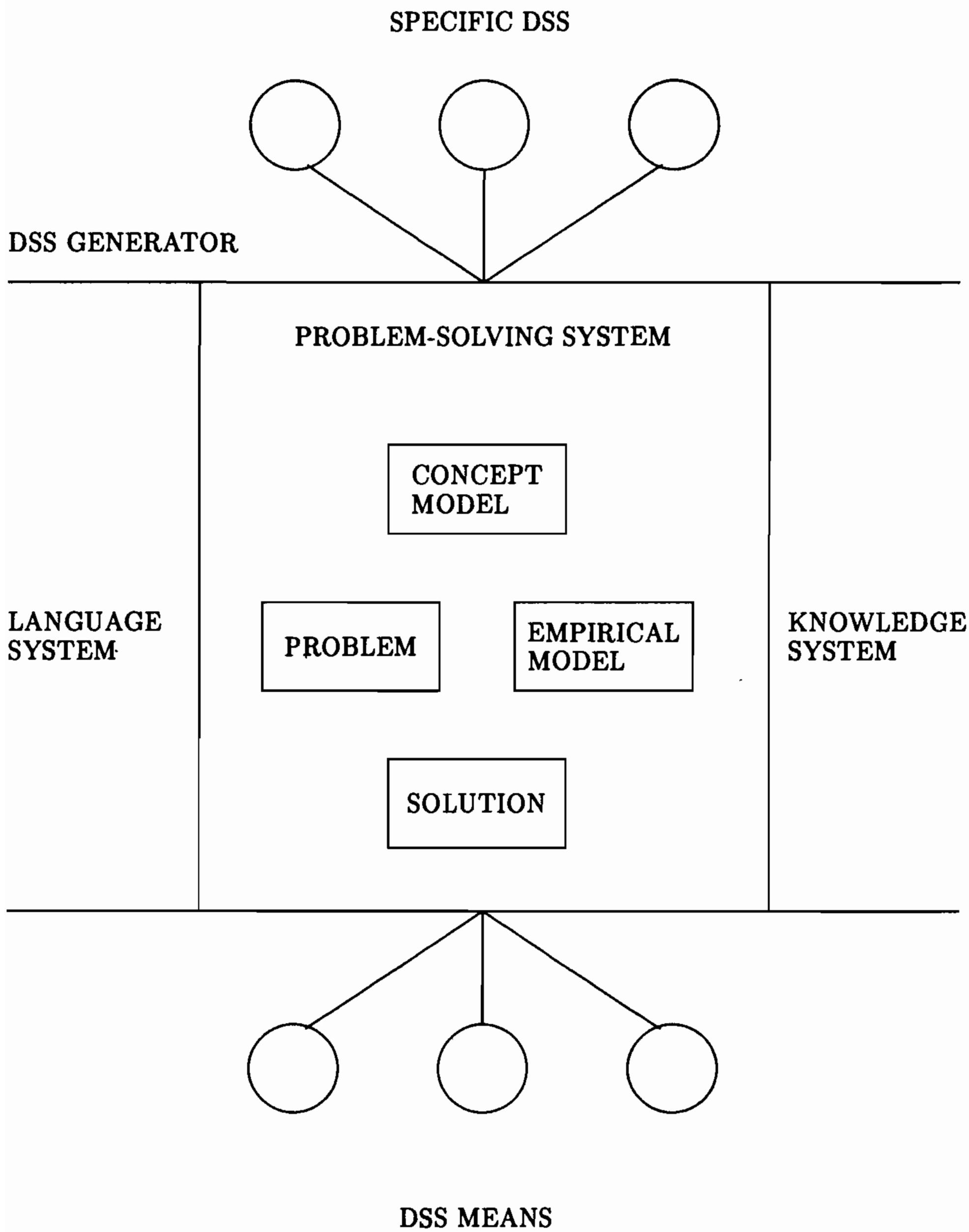

SPECIFIC DSS
DSS GENERATOR
PROBLEM-SOLVING SYSTEM
CONCEPT MODEL
LANGUAGE SYSTEM
PROBLEM
EMPIRICAL MODEL
KNOWLEDGE SYSTEM
SOLUTION
DSS MEANS

Figure 1

shaping the existing theories. In using the system in concept model building new theoretical concepts can arise;

- the results received during the different activities during problem — solving accumulate in the system as different levels of a multi-level structure. This makes easier their adaptation to individual theories as well as the system's evolution;

- the modelling environment of the reference system provides flexible support during all stages of decision-making.

The language system has a major role to play in the construction of conceptual and empirical models. It should not restrict those that search for and make decisions in making use of knowledge for model construction and proof accumulation. There is a difference between the descriptive form making use of equations and the processes and models constructed on the basis of rules.

Equation models are used often in DSS in strategic and organizational problems or in corporate and financial models. In such models functional links are defined as descriptive equations or state equations. Regardless of this, they can be represented through unconventional procedures, as it is done in some DSS generators.

In the processes or models based on rules, the process is not necessarily represented as an equation, it can be represented as a sequence of events. The concept "object" can be viewed as a defining multitude of connected attributes. An object can point to activities that could help alter the meaning of some attributes through one or more transformational rules or it can provide interaction with other objects. A script on a certain object can define the possible transformations as well as the ways of interaction between objects and the methods of their actualization. This can be done in different modes. One of these modes is the descriptive mode (Stamper, 1984), another one — the procedural mode that is implemented through procedures belonging to the programming languages. Regardless of the affirmations that these modes happen to be antagonistic and contradictory, the thruth is that actually they are in complementary distribution. The choice of this or that mode depends on the specific requirements of the users.

If one and the same object combines both data and activities then its representation should make use of object-oriented language (Cox, 1984).

The terms "object" and "script" enable us to disclose the so called "black box" and to unveil to each decision-maker the decisions for defining concepts and rules. In this way we can define the language system, the knowledge system and the system of problem-processing. In order to set up a conceptual model, the decision-maker would have to describe the method he'll use in actualizing the objects through their attributes. The script can describe the changes for each individual in the input mode and the decision-making in the problem solving. In describing the script the reactions of the decision-support system can suggest new transformational rules that have not been thought out up till then and even to new attributes and new interactions. The need to define the regional scope of activities of the user can arise as an object of the system. Such an object can be shaped as an attribute with a certain access to the other objects. The decision-maker keeps under observation only the given object, i.e. the attributes and a part of the script that have direct bearing on it. Of course this representation would change dynamically during the problem-solving process.

Through defining the process of different objects one can scan the dynamics of decision making and the behaviour that corresponds to it. In this way proofs for the decision-making are created under the given circumstances making use of the specific and up-to-date knowledge about the specific characteristic of the process. The reference system facilitates the process of defining the alternatives and the choosing of decisions. In step-by-step following of the stages of problem solving the user gets information not only about the solution of current problems but also about the possibility of any future problems that might crop up. The reference system can be considered as a problem-solving environment, and if computers are used, then it can be considered as a DSS environment.

The procedure of epistemological representation of a process can be facilitated by using an expert system on an existent knowledge base from the same range of activities. Besides, other expert systems can be used as well if their knowledge bases facilitate the process of problem solving. Expert systems applications:

- verifying the descriptive model;
- model evaluation and evaluation of the results of the experimental modelling phase;
- suggesting new alternatives.

When using a knowledge base, the following stages should be completed:

- the choice of suitable components for the systemic description of the existing situation;
- defining the objects of decision and description of the rules of the existing situation;
- developing simulation models and problem analysis;
- developing of the alternatives prototypes and the reaction during the experiment;
- transforming the prototype in a complete DSS.

In order to understand and evaluate the combination of an expert system, an interface of natural languages and a decision support system, one should have a clear idea about their respective parts in the process of decision-making. Usually the decision-maker has to choose between a number of variants and the greater the number, the more successful he could be and the lesser the probability of making a mistake becomes.

In critical situation decision-making one usually uses the accupast experience in solving similar situation problems. The practical base enables the decision-maker to define the relevant input data, and then quickly make the decision of implementing certain activities. The setting up of such a practical knowledge base, however, is a very long and labour-consuming process.

In order to simplify that process the system should meet the following requirements:

- the system can understand in the best possible way both the problem and the data, so that the data flow before the alternatives structuring phase would be shortened. In this way the decision-maker would be able to operate with the data and with the results of its processing;

- eliminating the language barrier.

The Decision Simulation Approach (DSIM) is the result of combining the conventional DSS with the operations research, data base control and the possibilities of artificial intelligence. Expert systems and natural languages interface facilitates the process of decision modelling in four basic aspects:

- facilitates the link between the input data and the processing algorithm or the display;

- provides a mechanism for describing the environment of non-procedural English-like languages;

- facilitates the defining of the most suitable algorithms or data sets;

- facilitates problem-solving in cases when the processing algorithm is insufficient or not suitable. Artificial intelligence creates special processing algorithms in such cases.

This means that nowadays DSS is often used in solving the "What if..." problem and the decision-modelling would give the answer to the "Why..." question. The better part of all DSS designers are satisfied with the standard software for the maintenance when in operation by its end-users and very often they themselves recommend this or that software package (Shoval, 1983b). Actually, the maintenance level of the DSS process can be evaluated from passive to normative.

Passive maintenance is the provision of the means only without any consideration as to the way they would be used or to the decisions that are to be made. The aim of such a DSS research is to focus attention on the software and the ways to stimulate further its development regardless of the problem of decision-making and raising of the efficiency. Passive maintenance gives the managers the working means that they need most since they are busy mainly with planning and administration. In this case DSS designers and consultants do not express their own opinion on the process of decision-making. No definite sphere is aimed at, the end-user works by himself, his workstation is turned into something that is similar to the telephone.

The traditional type of DSS are, in fact, automated assistants to the manager. The manager chooses an alternative and evaluates the result. Supposedly, the introduction of generation and analysis of the better part of the variants would contribute to raising the efficiency of decision-making. There is a conceptual, practical and reasonable proof to such an assumption. It gives greater possibilities to use analytical methods.

Useful but unnaplicable are many of the methods in management science. DSS emphasizes the need to activate them and gives the managements evaluation the upperhand, in this way lowering the need for aims specifications and multicriteria variants. The dogmatic, narrow view was popular only for a couple of years and it did not treat the problem of decision- and evaluation-quality.

DSS gradually were severed from management science and turned into passive systems supporting the calculating activities of the end-users. Some researches opposed this tendency but the greater number of specialists approved of this restricted usage of DSS.

DSS development needs the pragmatic approach to people, both decision-makers and those who use this type of technology, and not the unpractical, analytical methods. This means that the aims of management science still hold: to provide the means and regulations that will define normatively how decisions should be made and not to describe the ways they have been made up till now and the ways they have increases efficiency.

The trend defined by aim and normative maintenance bears directly on DSS — to use better analytical methods in making complex and important decisions. The argument that this approach is unpractical still holds but not completely. Theory must have the major role but it must summarize the experience from a sequence of research operations.

The extension to the process of decision-making is an intermediary link between traditional and normative DSS maintenance. It focuses attention on the ways to influence the manager's decisions, his point of view with regard to using specific analytical means, etc.

In the extension to the process of decision-making one can define the following characteristics:

- spheres of possible DSS applications for new, unexpected activities;
- striving after the application of analytical methods and model decision analysis, decision-making on the basis of multicriteria and management models;
- The extended maintenance includes a better understanding of the operational capabilities of DSS. She restores many of the initial aims of DSS i.e. it influences the quality of the decisions that are made.
- making use of new programming means and artificial intelligence knowledge bases;
- service both to intellectual and information technology;
- making use of the DSS designer's knowledge in a specific field in defining and choosing the various decision variants and not just in evaluating the different possibilities and decision choice.

Researchers and designers of the descriptive model have focused their attention on the separate decision-maker and his psychology. In fact, very little attention was devoted to group decision-making or on decision-making within the organization as a whole. As a result there is a trend in DSS to support the decision of the separate manager only who is to make a decision on BL and not decisions on the budget of the whole organization.

The autonomous technology of decision-making on separate tasks raises individual productivity but is concerned only with specific restricted problems. Telecommunications, data storage and company data alter the very essence of DSS as well as its potentials.

DSS combine together both practical and research activities. Practical activities should influence research, and research, in its turn, should be directed at perfecting practice, so that the major task would be fulfilled — to offer the manager the help he needs. The intelligence of maintenance is something completely different from research. Intelligence requires knowledge in other spheres, usually new ones, that form the basis of research activities. The importance of the "intelligence" of maintenance is connected to the critical choice among a great number of sources and the system of knowledge within them, especially where multicriteria decision-making and finance theory are concerned. An intelligent control on the quality of knowledge must be exercised.

Now we can define two trends in DSS development — integration and application of the artificial intelligence means. While artificial intelligence attempts at developing new means and tools that would probably have small bearing on final effects, integration can be achieved through means that are already in existence, and in this case effects would be fast and easily defined. The continuous efforts at analyzing and modelling decision-making are aimed at setting up a basis for the further automatization of integration.

ES applications in decision-making are not restricted to diagnostics in medicine or equipment servicing. Just the opposite, ES can be used in any situation that requires some type of judgment. For example, ES can partly substitute the decisions of the financial expert who reviews the financial practice of an organization with giving him objective standards that do not depend on one's mood and predisposition (Hansen and Messier, 1982).

ES–DSS integration would be further stimulated. On the one hand, ES and the possibilities of natural languages will make the administrators use DSS modelling capabilities as well as the quantitative characteristics of ES and their capabilities to deduce judgments (Turban and Watkins, 1986). On the other hand, mass application of modelling would lead to further elimination of many experts spheres and this, in its turn, may lead to the development of new ES.

ES–DSS integration can be implemented on all three levels of DSS: data base, model base and dialogue subsystem. Table 1 shows that the links are in both directions, i.e. DSS can increase ES productivity and vice versa. Besides, DSS and ES can be integrated as two systems complementing each other (for example DSS output is ES input).

In conclusion we can define that the answers to the following questions would determine the progress in DSS research:

- which decisions are the most important for the organization and what means the decision-maker would need to make the right decision;

- how to organize specialists' introduction? How to design DSS means that would perfect the dialogue, that would stimulate instruction in its turn?

- what is the role of modelling in developing creative thinking;

- what technical architecture is needed for DSS development from the point of view of the ever-complicated technologies;

- how can a researcher evaluate efficiency and the quality of the decisions that are made.

These problems have a direct bearing on our understanding of decision-making from the following points of view:

- the application and accessibility of computers is raising the efficiency in solving ill-structured problems;

Subsystem	Links Description
Data base	• ES perfects the functioning, development and maintenance of the data base • the data base can provide information for ES; • access is perfected through natural language subsystems; • DSS can set up a data base for ES and dialogue with the users.
Model base	• users experience can help in choosing the model; • ES can perfect the model control; • ES can perfect the analysis programs and the system sensitivity; • DSS can make analysis for ES (prognostics for example) • ES can provide euristics; • DSS can provide facts for ES; • ES can help research on simulation models; • ES can use quantitative models.
Dialogue subsystem	• Artificial intelligence complements the natural language processor; • ES provides symbolic languages that guarantee the interface user-friendliness; • ES gives new means for explanation; • ES provides terms and references.

Table 1

- making use of intelligence and computer technologies for perfecting creative activities and important decision- making.

Practical activities should include:

- the choice of active and not passive or undefined activities;
- support to the key process of the organization and not just solving ad-hoc or individual problems;
- setting up of a stable organizational environment defined by the users and the level of maintenance they require; setting of priorities from the point of view of economic efficiency and the achievements of the knowledge sources, etc.;
- making use of new means, technologies, documents, telecommunication systems;
- combining artificial intelligence means with DSS generation means.

References

Cats-Baril, W. and Huber, G.P. (1984). DSS for Ill-structured Problems : a Cognitive Approach and an Empirical Study. Proceedings of the IFIP 8.3 Working Conference on Knowledge Representation for DSS.

Chen Ye-Sho, (1988). An Entity — Relationship Approach to Decision Support and Expert Systems. *Decision Support Systems Journal*, June 1988, Vol. 4, No. 2.

Cox, C.Y. (1984). Message/Object Programming: An Evolutionary Change in Programming Technology. IEEE Software.

Fordyce, G.S. (1985). Examples of the Role of Expert Systems in Decision Support. IEEE.

Gevarter, W. (1983). Expert Systems: Limited but Powerful. *LEEE Spectrum* 8.

Hansen, J. and Messier, W. (1982). Expert Systems for Decision Support in EDP auditive. *Internal J. comput. and info. sci.*.

Keen, P.G.W. (1983). Adaptive Design for Decision Support Systems. *Data Base* 12.

Martins Gary, R. (1984). The Overselling of Expert Systems. Datamation, November 1st, 1984, see also Smoke and Mirrors?, Unix Review, August, 1987.

Shoval, P. (1983a). Knowledge Representation in Consultation Systems for Users of Retrieval Systems. The Application of Mini and Macro Computers in Information Documentation and Libraries, the Netherlands, North-Holland.

Shoval, P. (1983b). Knowledge Representation in Consultation Systems for Users of Retrieval of Information. Documentation and Libraries, the Netherlands, North-Holland.

Sol, H.G. (1985). Aggregating Data for Decision Support. *Decision Support Systems*, No. 2.

Stamper, R. (1984). Management Epistemology: Garbage In, Garbage Out. Proceedings of the IFIP 8.3 Working Conference on Knowledge Representation for DSS.

Turban, E. and Watkins, P. (1986). Integrating Expert Systems and Decision Support Systems. *MIS Quarterly*, June 1986.

William, C.H. (1984). Decision Support Systems. Petrocelli Books.

Multicriterion Multiextreme Optimization with Nonlinear Constraints

R.G. Strongin, V.P. Gergel, D.L. Markin
Gorky State University
Gorky, USSR

Abstract

In this paper there is described the method of solving the multidimension multiextreme multicriterion optimization problems with complex constraints that are typical mathematical models of procedures used in selecting optimal decisions for computer-aided design of objects and processes (Podinovsky and Nogin, 1982; Evtoushenko, 1982; Mikhalevith and Volkovich, 1982).

Proposed below are the formal description of a general problem of multiextreme optimization,the algorithm for its solution and the results of the test example computation.

1 The general multiextreme problem

In the N-dimensional hyperinterval

$$D = \{\, y \in R^n : a_i \leq y_i \leq b_i,\ 1 \leq i \leq N \,\} \tag{1}$$

there is defined such a vector-function

$$w(y) = (w_1(y), \ldots, w_n(y)), \quad w_i(y) > 0, \quad 1 \leq i \leq n \tag{2}$$

that reducing each coordinate function $w_i(y)$ will facilitate to efficiently solve the general problem. One portion of coordinate functions $w_i(y)$ from (2) will make up a vector efficiency criterion

$$f(y) = (f_1(y), \ldots, f_n(y)). \tag{3}$$

Each specific criterion $f_i(y)$ is required to be reduced to the most extent possible.

The coordinate functions $w_i(y)$ from (2), not included into the efficiency criterion (3), have to be minimized in order to fulfill the inequalities

$$w_{i_j}(y) \leq q_j, \quad 1 \leq j \leq m, \quad m = n - k, \tag{4}$$

where the values $q_j > 0, \ 1 \leq j \leq m$, have been declared and constitute a vector of constraints. With the help of inequalities (4) we distinguish some subset y of valid solutions from the set of all solutions and define the constraining vector

$$g(y) = (g_1(y), \ldots, g_m(y)) = (w_{i_1}(y) - q_1, \ldots, w_{i_m}(y) - q_m) \tag{5}$$

the nonpositiveness of the components of which constitutes an essential and sufficient condition for the possibility of solution y.

Hence we set now a problem of minimizing the vector efficiency criterion $f(y)$ from (3) within the subset of valid points

$$Q = \{ y \in D : g_i(y) \leq 0,\ 1 \leq i \leq m \}, \quad \min \{ f(y) : y \in Q \} \tag{6}$$

The efficient (not improved) solutions (Podinovsky and Nogin, 1982) are considered to be a solution of the problem (6).

Through the penalty functions (see this method in Krasnoshekov et al., 1979; Germeier, 1971), the estimation of some specific efficient solution can be reduced to a one-step problem

$$y^* = y^*_\lambda(\beta) = \arg \min_{y \in Q} \Big\{ \max_{1 \leq i \leq k} (\lambda_i f_i(y)) + \beta \sum_{i=1}^{k} f_i(y) \Big\} \tag{7}$$

for a given weighting vector λ and sufficiently small positive ratio β. So in the capacity of some approximate solution of the problem (6) we shall consider a set $P_\delta(Q)$ obtained upon solving the problem (7) for each vector λ from δ-grid $\Lambda_\delta \subset \Lambda$ where

$$\begin{aligned} &\Lambda = \Big\{ \lambda \in R^k : \sum_{i=1}^{\kappa} \lambda_i = 1,\ \lambda_i \geq 0,\ 1 \leq i \leq k \Big\}, \\ &\delta = (\delta_1, \ldots, \delta_k), \quad 0 < \delta \leq 1, \quad 1 \leq i \leq k. \end{aligned} \tag{8}$$

2 The method of solving the general multiextreme problem

Under the abovesaid, estimating the solution of a general multidimension multiextreme multicriterion problem with constraints (6) is reduced to solving a series of multiextreme optimization problems with nonlinear nonconvex constraints

$$\begin{aligned} &\min \{ \varphi(y) : y \in D,\ g_i(y) \leq 0,\ 1 \leq i \leq m \Big\}, \\ &\varphi(y) = \max_{1 \leq i \leq k} \lambda_i f_i(y) + \beta \sum_{i=1}^{k} f_i(y), \end{aligned} \tag{9}$$

where D is from (1), $f_i(y)$ and $g_i(y)$ are from (7) and (3), and weighting vector λ belongs to the simplex Λ from (8), and the ratio β ($\beta > 0$) has to be sufficiently small. Below described are one of the techniques for reducing the problem (9) to a one-dimension problem with constraints and a numerical method for solving such one-dimension problems without use of penalty functions. The approach we have adopted can be substantiated by the following:

a) the considered by us global search algorithm for the optimized function satisfying the Lipshits condition will generate a minimizing sequence converging only to the optimal points (Strongin and Markin, 1985, 1986);

b) in minimizing the functional (9), if a provision is made for storing all parameters $w^\nu = w(y^\nu)$ associated with the values of vector-function $w(y)$ from (2) that are computed in the points y^ν, $0 \le \nu \le k$, then when modifying the weighting vector to estimate a next efficient solution it will become possible to compute the value $z^\nu = \varphi_\lambda(y^\nu)$ of the new functional (9) in the same points y^ν without turning to a labour-consuming computation of the parameters w^ν. At this, the currently employed algorithm will start to solve a new problem immediately from the k-th step and this undoubtedly will bring a speedup in the problem solution (see the results brought by the test example).

2.1 Reducing to the one-dimension problem

A segment $[0,1]$ of some real axis x can be uniquely and continuously mapped into an N-dimension hiperinterval D from (1). The mappings of this kind are called Peano space filling curves or Peano curves. If the minimizing function $\varphi(y)$ from (9) is continuous, then the continuity of the curve $y(x)$ will make valid the following equality

$$\min_{y \in D} \varphi(y) = \min_{x \in [0,1]} \varphi(y(x))$$

and the multidimension problem of minimizing the function $\varphi(y)$ will then be reduced to that of minimizing the one-dimension function $\varphi(y(x))$; at this if the function $\varphi(y)$ being of a Lipshits type, then the function $\varphi(y(x))$ will meet the the Gelder condition with index N^{-1}. Besides, the correspondence $y(x)$ is not of a one-to-one type and hence the point $y \in D$ at this correspondence can obtain several (up to 2^N) preimages. The algorithms for approximate computations of images $y(x)$ and preimages of points $y \in D$ are discussed in (Strongin, 1978).

2.2 The algorithm for solving the one-dimension problem

According to the Section 2.1 the multidimension optimization problem (9) is reduced to the following one-dimension problem

$$\min \left\{ \varphi(y(x)) : x \in [0,1],\ g_i(y(x)) \le 0,\ 1 \le i \le m \right\} \tag{10}$$

and besides the minimizing function $\varphi(y(x))$ (denoted later as $g_{m+1}(y(x))$) and left parts of constraints $g_i(y(x))$, $1 \le i \le m$, are supposed to be defined only in the associated subareas Q_i

$$Q_0 = [0,1], \quad Q_{i+1} = \left\{ x \in Q_i : g_i(y(x)) \le 0 \right\}, \quad 1 \le i \le m.$$

The latter is specific for the problems of optimal design when a failure in any constraints results in the fact that series of features of the object under optimization will become not sufficiently defined.

The points of the segment $[0,1]$ are classified due to the number of constraints executed in them, and this classification is done with the index $\nu = \nu(x)$, $1 \le \nu \le m+1$ described by the conditions $x \in Q_\nu$ and $x \notin Q_{\nu+1}$ (it is assumed that $Q_{m+2} = \{\emptyset\}$). The maximum value of index M is introduced and this can serve some kind of indicator of noncompatibility for the problem constraints $M < m+1$ (10).

The first iteration of the algorithm is performed in some arbitrary point $x^1 \in (0,1)$. The point x^{k+1}, $k \geq 1$, is selected in accordance with the following rules:

1) the points $x^1, \dots, x^k$ of previous iterations are numbered by subindices according to the ascending values of the coordinate, i.e.

$$0 = x_0 < x_1 < \ldots < x_i < \ldots < x_k < x_{k+1} = 1 \tag{11}$$

and each point x_i, $1 \leq i \leq k$, is matched against the value $z_i = g_\nu(y(x_i))$ computed in it, where $\nu = \nu(x_i)$ (the points $x_0 = 0$ and they are additionally introduced and formally interpreted as possessing a zero index; the values z_0 and z_{k+1} have no definition and at this each new point x^{k+1} is put into the series (11) together with its preimages and they all are assigned one and the same value z^{k+1};

2) there are computed the parameters

$$\mu_\nu = \max\{\, |z_i - z_j| / \rho_{ij} : i,j \in I_\nu,\ i > j\,\}, \quad 1 \leq \nu \leq m+1,$$
$$z_\nu^* = \begin{cases} 0 & , \ T_\nu \neq \{\emptyset\} \\ \min\{\, z_i : i \in I_\nu \,\} & , \ T_\nu = \{\emptyset\} \end{cases}$$

where

$$I_\nu = \{\, i : 1 \leq i \leq k,\ \nu = \nu(x_i)\,\}, \quad T_\nu = I_{\nu+1} \cup \ldots \cup I_{m+1}, \quad \rho_{ij} = |x_i - x_j|^{1/N},$$

and at this for $|I_\nu| < 2$ or $\mu_\nu = 0$ it is supposed that $\mu_\nu = 1$;

3) for each interval (x_{i-1}, x_i), $1 \leq i \leq k+1$, we computed the feature

$$R(i) = \begin{cases} \left\{\rho_{i,i-1}\mu_\nu + \dfrac{(z_i - z_{i-1})^2}{\rho_{i,i-1}\mu_\nu} - 2\,\dfrac{z_i + z_{i-1} - 2z_\nu^*}{z}\right\} \Big/ \\ \qquad \left\{\left((z_i - z_\nu^*)(z_{i-1} - z_\nu^*)\right)^{1/N} + \alpha\mu_\nu\right\}, \qquad \nu = \nu(x_i) = \nu(x_{i-1}), \\ \left\{2\rho_{i,i-1}\mu_\nu - 4\,\dfrac{z_{i-1} - z_\nu^*}{z}\right\} \Big/ \left\{(z_{i-1} - z_\nu^*)^{2/N} + \alpha\mu_\nu\right\}, \\ \qquad \nu = \nu(x_{i-1}) > \nu(x_i), \\ \left\{2\rho_{i,i-1}\mu_\nu - 4\,\dfrac{z_i - z_\nu^*}{z}\right\} \Big/ \left\{(z_i - z_\nu^*)^{2/N} + \alpha\mu_\nu\right\}, \\ \qquad \nu = \nu(x_i) > \nu(x_{i-1}), \end{cases}$$

where z $(z > 1)$ and α $(0 < \alpha < 1)$ are parameters of the algorithm;

4) it is assumed that

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - \begin{cases} 0 & , \quad \nu(x_t) \neq \nu(x_{t-1}) \\ \dfrac{\operatorname{sign}(z_t - z_{t-1})}{2z} \left\{ \dfrac{|z_t - z_{t-1}|}{\mu_\nu} \right\}^N & , \quad \nu = \nu(x_t) = \nu(x_{t-1}) \end{cases}$$

where $t = \arg\max \{R(i) : 1 \leq i \leq k+1\}$.

The halting condition will terminate the computation either when overlapping of the preimages occurs or upon a present number of executed iterations or upon approaching some given accuracy ε, i.e. upon solving the inequality $\rho_{t,t-1} \leq \varepsilon$.

If the minimizing function φ from (9) is a sufficiently smooth one (that occurs when all the computations constituting the base for the estimate φ are exercised with sufficient accuracy), then it is good to combine global iterations with local search. This is well regulated with the help of the parameter α, and so $\alpha \sim 1$ will denote a global search and $\alpha \sim 0$ a local search. It should be noted that search data is acquired in any iterations of global and local searches.

3 The test example

The test for a general multiextreme problem is of the following kind:

- the vector criterion is $f(y) = (f_1(y), f_2(y))$, where

$$\begin{aligned} f_1(y) &= \left(100(y_2 - (y_1 - 2)^2)^2 + (3 - y_1)^2\right)^{1/2}, \\ f_2(y) &= 1.5 - 1.5y_1^2 \exp[1 - y_1^2 - 20.25(y_1 - y_2)^2] + \\ &\quad + (0.5y_1 - 0.5)^4 \cdot (y_2 - 1)^4 \exp[2 - (0.5y_1 - 0.5)^4 - (y_2 - 1)^4]; \end{aligned}$$

- the constraint vector is $g(y) = (g_1(y), g_2(y), g_3(y))$, where

$$\begin{aligned} g_1(y) &= (y_1 - 2.2)^2 + (y_2 - 1.2)^2 - 2.25 \leq 0, \\ g_2(y) &= 1 - ((y_1 - 2)/1.2)^2 - (y_2/2)^2 \leq 0, \\ g_3(y) &= y_2 - \sin(2y_1 + 0.25) - 1 \leq 0, \end{aligned}$$

- the search range is $D = \{\, y \in R^2 : 0 \leq y_1 \leq 4, \ -1 \leq y_2 \leq 3 \,\}$.

The results of solving the test example are given in the table, where (λ_1, λ_2) is a weighting vector from (8); k_1 is the number of iterations for global search upon which the below combinations has been applied: one step includes a global search and another step a local search; k_2 denotes a number of steps prior to the halt. The Figure 1 shows the results of solving the first problem, given on it are the boundaries of admitted area, the lines of constant level for the first criterion and the points in which functional $\varphi(y)$ was computed (here the points falling into the admitted area are denoted with character "x"). The corresponding data for the problem 4 is presented in Figure 2, on it only new points of trials are given whereas in the problem 1–3 the optimizing.

No.	The weighting vector λ		The number of the iteration for initiatiting the local steps	The amount of search iterations needed
	λ_1	λ_2	k_1	k_2
1	1.0	0.0	100	200
2	0.0	1.0	80	117
3	0.25	0.75	0	120
4	0.5	0.5	0	94
5	0.75	0.25	0	1
Total number of points				532
Efficient point				25

References

Evtoushenko, Ju.G. (1982). The methods for solving extreme problems and their application in optimization systems. Moskow, Nauka, (in Russian).

Germeier, Ju.B. (1971). Introduction into the theory of operation research. Moskow, Nauka, (in Russian).

Krasnoshekov, P.S., Morozov, V.V., Fedorov, V.V. (1979). The decomposition in the problems of design 1. Izv. AN USSR, *Technicheskaja kibernetica*, No. 2, pp. 7–17, (in Russian).

Mikhalevith, V.B. and Volkovich, V.L. (1982). Computational methods of research and designing complex systems. Moskow, Nauka, (in Russian).

Podinovsky, V.V. and Nogin V.D. (1982). Pareto-optimal solutions for multicriterion problems. Moskow, Nauka, (in Russian).

Strongin, R.G. (1978). Numerical methods in multiextreme problems (Information and statistical algorithms). Moskow, Nauka, (in Russian).

Strongin, R.G. and Markin, D.L. (1985). Global optimization with use of apriori distribution. In: VII All-Union Conference "Problemy teoreticheskoy kibernetiki", report abstracts, part 2, Irkoutsk, (in Russian).

Strongin, R.G. and Markin, D.L. (1986). Minimizing the multiextreme functions at non convex constraints. *Kibernetica*, No. 2, (in Russian).

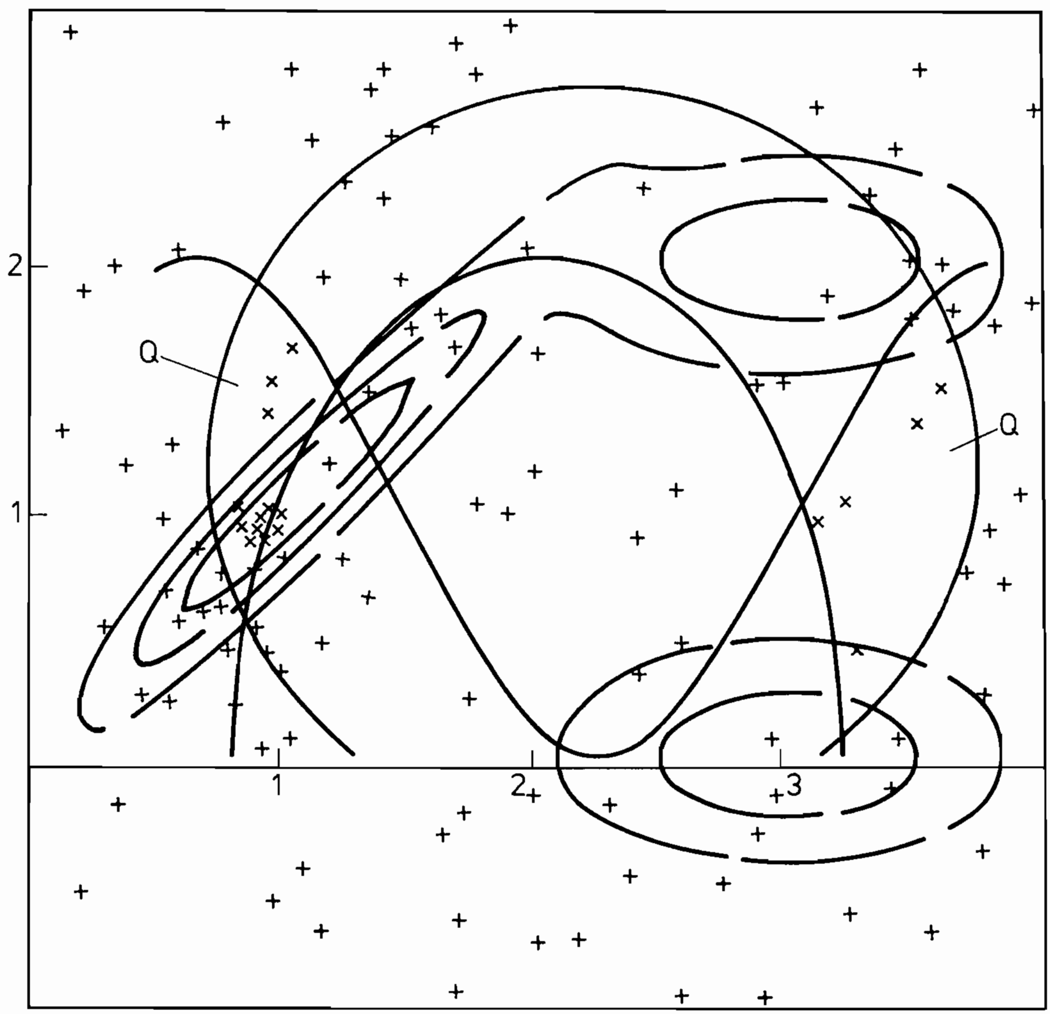


Figure 1: Distribution of points of trials while minimizing the first criterion ($\lambda = (1, 0)$).

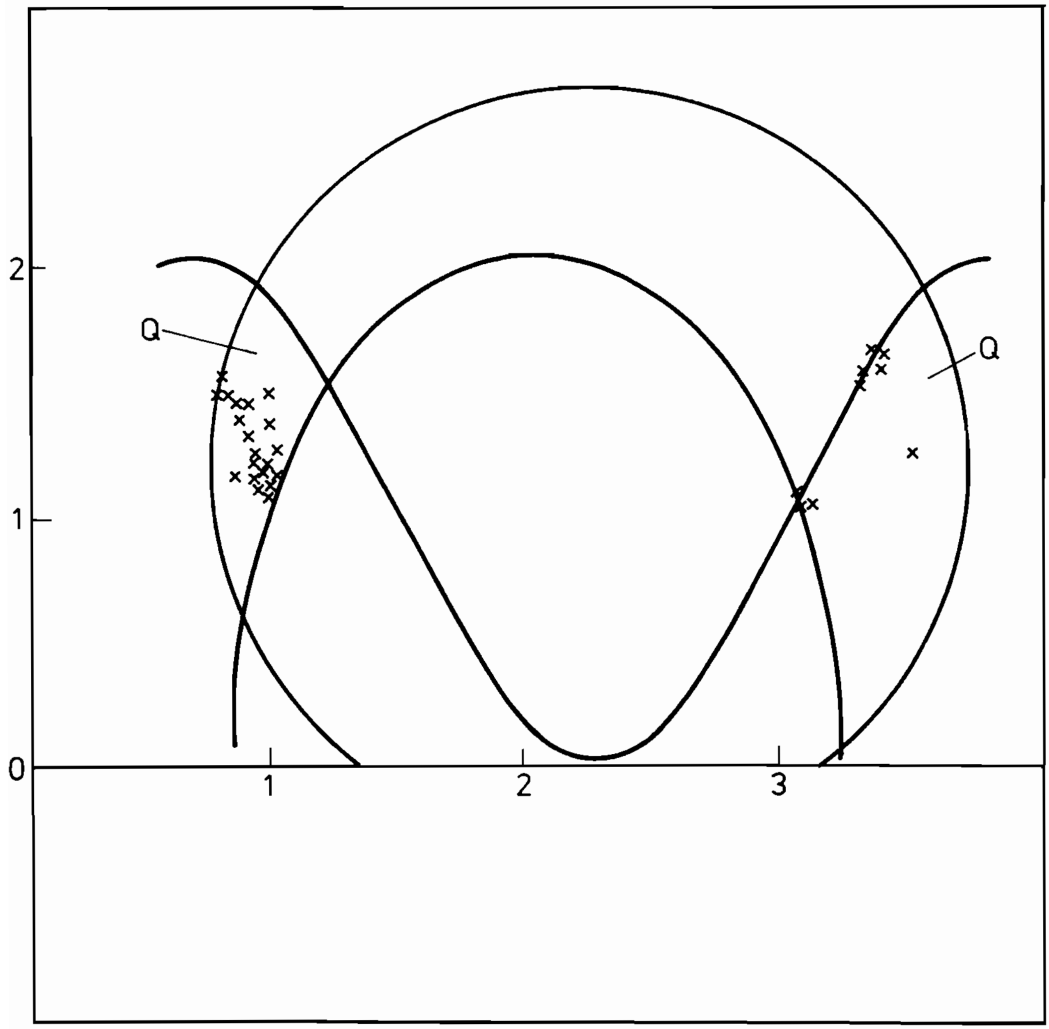


Figure 2: Distribution of points of trials while evaluating of effective solution for $\lambda = (0.5, 0.5)$.

Optimization Methods for Two-Level Multiobjective Problems

Tetsuzo Tanino
Department of Information Technology
Okayama University, Okayama 700, Japan

Harukazu Satomi
Department of Mechanical Engineering II
Tohoku University, Sendai 980, Japan

1 Introduction

In this paper we propose optimization methods for two-level programming problems (with multiple objectives). Recently we must often deal with large scale optimization problems in many fields of engineering and economics. In some cases those problems are formulated as (typically two-level) hierarchical problems (Mesarovic et al., 1970). Though several types of hierarchical optimization problems may be considered, we concentrate on those in which the optimal value functions are included in the upper level objective functions. Some authors have provided optimization methods for those problems. For example, Shimizu and Ishizuka (1985) proposed a method based on the non-differentiable optimization method by Mifflin (1977) under the linear independence constraint qualification in the lower level optimization. Tanino and Ogawa (1984) proposed a method effective for convex hierarchical problems.

In any cases we require strong conditions such as convexity or linear independence constraint qualification. Hence their applicability is fairly limited. In this paper we propose a method which is valid under a milder condition — Cottle constraint qualification. Moreover the method is shown to be applicable to a class of multiobjective two-level optimization problems. (In this case, we suppose that the optimal set is a singleton).

The contents of this paper are as follows. First, several useful results in sensitivity analysis in nonlinear programming are reviewed. Based on those results, steepest descent direction finding problems in the upper level optimization problems are formulated as simple (for example, quadratic) programming problems. A practical algorithm for solving two-level optimization problems are provided. The results are extended to two-level multiobjective problems.

2 Sensitivity analysis in nonlinear programming

In this section we review some results about sensitivity analysis in nonlinear programming. A nonlinear programming problem considered here is as follows:

$$
\begin{array}{lll}
(\mathrm{P}(u)) & \underset{x}{\text{minimize}} & f(x,u) \\
& \text{subject to} & g_j(x,u) \le 0, \qquad j = 1, \dots, q
\end{array}
$$

where $u \in R^m$ is a perturbation parameter vector, $x \in R^n$ is a decision vector, $f : R^n \times R^m \to R$ is an objective function and $g_j : R^n \times R^m \to R$ $(j = 1, \dots, g)$ is a constraint function.

The optimal value function for this problem is defined by

$$f^*(u) = \inf\{f(x,u) \mid g_j(x,u) \le 0, \quad j = 1, \dots, q\}$$

We denote the set of optimal solutions to this problem by

$$S(u) = \{\hat{x} \mid f(\hat{x},u) = f^*(u), \quad g_j(\hat{x},u) \le 0, \quad j = 1, \dots, q\}$$

and the set of indices of active constraints by

$$J(x,u) = \{j \mid g_j(x,u) = 0, \quad j = 1, \dots, q\}$$

The Lagrange function for the problem is defined by

$$L(x,u,\lambda) = f(x,u) \sum_{j=1}^{q} \lambda_j g_j(x,u)$$

The set of the Lagrange multiplier vectors which satisfy the Kuhn-Tucker conditions is defined and denoted by

$$K(x,u) = \{\lambda \mid \nabla_x L(x,u,\lambda) = 0, \quad \lambda_j \ge 0, \quad \lambda_j = 0 \quad j \notin J(x,u)\}.$$

2.1 The case where gradients can be obtained

We have the following theorem concerning the possibility of obtaining the gradient vector of the optimal value function.

<u>Theorem 2.1.</u> (Fiacco, 1976 and 1983) Assume the following in Problem $(\mathrm{P}(u))$.

1. The functions f and g_j $(j = 1, \dots, q)$ are all twice continuously differentiable;
2. The set of feasible solutions is compact for every u is some neighborhood of $\hat{u}$;
3. The set of optimal solutions $S(\hat{u})$ consists of a unique element $\hat{x}$;
4. $\{\nabla_x g_j(\hat{x},\hat{u}), \quad j \in J(\hat{x},\hat{u})\}$ are linearly independent;
5. The strict complementary slackness holds at $\hat{x}$, i.e., $J = \tilde{J}$, where

$$\tilde{J} = \{j \mid \lambda_j > 0, \quad j = 1, \dots, q\};$$

6. The second order sufficiency conditions hold at $\hat{x}$.

Then f^* is differentiable at $\hat{u}$ and

$$\nabla_u f^*(\hat{u}) = \nabla_u f(\hat{x}, \hat{u}) + \sum_{j=1}^{q} \hat{\lambda}_j \nabla_u g_j(\hat{x}, \hat{u})$$

where $\hat{\lambda}_j$ is the Lagrange multiplier corresponding to $\hat{x}$.

This theorem implies that we may obtain the gradient vector of the optimal value function if the optimal solution is unique and if the linear independence constraint qualification holds.

2.2 The case where generalized gradients can be obtained

We can obtain the generalized gradient of the optimal value function if the conditions in the following theorem are satisfied.

Theorem 2.2. (Gauvin & Dubeau, 1982) Assume the following in Problem (P(u)).

1. The functions f and g_j ($j = 1, \ldots, q$) are all twice continuously differentiable;
2. The set of feasible solutions is uniformly compact near $\hat{u}$;
3. $\{\nabla_x g_j(\hat{x}, \hat{u}), \quad j \in J(\hat{x}, \hat{u})\}$ are linearly independent for every $\hat{x} \in S(\hat{u})$;

Then the generalized gradient of f^* at $\hat{u}$ is given by

$$\partial^0 f^*(\hat{u}) = co\{ \bigcup_{\hat{x} \in S(\hat{u})} \nabla_u L(\hat{x}, \hat{u}, \hat{\lambda})\}$$

Theorem 2.2 shows that we may obtain the generalized gradient if the linear independence constraint qualification holds, even if the optimal solution is not unique.

2.3 The case where directional derivatives can be obtained

As can be seen from Theorems 2.1 and 2.2, if the linear independence constraint qualification does not hold, i.e., the Lagrange multiplier vector is not unique, it is not generally possible to obtain the gradient or the generalized gradient value function. However, even in those cases, if the Cottle constraint qualification holds, we may obtain the directional derivatives. Here the Cottle constraint qualification is said to hold at $\hat{x}$ if there exists $h \in R^n$ such that

$$\langle \nabla_x g_j(\hat{x}, \hat{u}), h \rangle < 0 \qquad \forall j \in J(\hat{x}, \hat{u}).$$

Here $\langle , \rangle$ denotes the inner product in the Euclidean spaces.

Theorem 2.3. (Rockafellar, 1984) Assume the following in Problem (P(u)).

1. The functions f and g_j $(j = 1, \ldots, q)$ are all twice continuously differentiable;
2. Inf-boundedness assumption is satisfied, and the set of feasible solutions is not empty;
3. The Cottle constraint qualification holds at every optimal solution $\hat{x} \in S(\hat{u})$.
4. $ri\, K(\hat{x}, \hat{u}) \subset K^a(\hat{u})$, where $K^a(\hat{u})$ is the set of multiplier vectors $\hat{\lambda}$ such that for a sufficiently large penalty parameter, $(\hat{x}, \hat{\lambda})$ is a local saddle point of the augmented Lagrangian for $u = \hat{u}$.

Then f^* possesses finite one-sided directional derivatives at $\hat{u}$ in the Hadamard sense. In fact, for every v

$$f*'(\hat{u}; v) = \min_{x \in S(\hat{u})} \max_{\lambda \in K(x, \hat{u})} \langle \nabla_u L(x, \hat{u}, \lambda), v \rangle$$

Thus the results about sensitivity of the optimal value function depend strongly on the uniqueness of the optimal solution and the Lagrange multiplier vector. This fact is shown in Table 1.

Case (1): The gradient of the optimal value function is available.

Case (3): The generalized gradient is available. However it is not generally possible to obtain the complete set of the optimal solutions.

Case (2): The directional derivatives are available.

Of course some additional conditions are necessary in each case.

		Lagrange multipliers	
		unique	set
optimal	unique	(1)	(2)
solutions	set	(3)	

Table 1: Classification by optimal solutions and Lagrange multipliers

3 Derivatives of the upper level objective function

Now we deal with the following two-level optimization problem:

$$(T) \qquad \underset{u}{\text{minimize}} \quad \phi(u) = \psi(u, f^*(u))$$

where

$$f^*(u) = \inf\{f(x, u) \mid g_j(x, u) \leq 0, \quad j = 1, \ldots, q\}$$

and $u \in R^m$ is the upper level decision vector, $x \in R^n$ is the lower level decision vector, $\psi: \; R^m \times R \to R$ is the upper level objective function, $f(x, u): \; R^n \times R^m \to R$ is the lower level objective function and $g_j(x,u)$ $(j = 1, \ldots, q)$: $R^n \times R^m \to R$ is the lower level inequality constraint.

In view of Theorem 2.1, if the optimal solution is unique and the linear independence constraint qualification holds, then the optimal value function is differentiable under some additional conditions. Hence, if we assume that ψ is continuously differentiable in (u, f),

$$\nabla_u \psi(\hat{u}, f^*(\hat{u})) + \nabla_f \psi(\hat{u}, f^*(\hat{u})) \cdot \nabla_u f^*(\hat{u})$$

If the linear independence constraint qualification does not hold, but if the Cottle constraint qualification holds, then the optimal value function is directionally differentiable as was shown in Theorem 2.3. We discuss the latter case in detail.

First we note the following lemma concerning the set of Lagrange multiplier vectors.

Lemma 3.1. (Gauvin and Tolle, 1977) Suppose that Problem (P($\hat{u}$)) has a solution $\hat{x}$. Then the set of Lagrange multiplier vectors $K(\hat{x}, \hat{u})$ is a nonempty, compact, and convex set if and only if the Cottle constraint qualification holds at $\hat{x}$.

Hereafter in this section we assume that Problem (P($\hat{u}$)) has a unique solution $\hat{x}$. In order to compute directional derivatives, we only need the values of the elements of the Lagrange multiplier vector λ corresponding to the active constraints. Hence we denote this subvector of λ by λ_J. Let

$$K_J(\hat{x}, \hat{u}) = \Big\{\lambda_J \mid \nabla_x f(\hat{x}, \hat{u}) + \sum_{j \in J(\hat{x}, \hat{u})} \lambda_j \nabla_x g_j(\hat{x}, \hat{u}) = 0, \quad \lambda_j \geq 0\Big\}.$$

and $G(\hat{x}, \hat{u})$ be an $n \times \#J$ matrix consisting of the column vectors $\nabla_x g_j(\hat{x}, \hat{u})$ $(j \in J(\hat{x}, \hat{u}))$, where $\#J$ is the number of the elements in J. Then $K_J(\hat{x}, \hat{u})$ can be rewritten as follows:

$$K_J(\hat{x}, \hat{u}) = \{\lambda_J \mid \lambda_J \geq 0, \quad G\lambda_J = -\nabla_x f(\hat{x}, \hat{u})\}$$

If the Cottle constraint qualification holds at $(\hat{x}, \hat{u})$, $K_J(\hat{x}, \hat{u})$ is a nonempty, compact and convex set from Lemma 3.1. Hence, by Theorem 2.3, the directional derivative of f^* at $\hat{u}$ in the direction v is given by

$$\begin{aligned} f'^*(\hat{u}; v) &= \max_{\lambda_J \in K_J(\hat{x}, \hat{u})} \Big\{\langle \nabla_u f(\hat{x}, \hat{u}), v\rangle + \sum_{j \in J(x, \hat{u})} \lambda_j \nabla_u g_j(\hat{x}, \hat{u}), v\rangle\Big\} = \\ &= \max_{\lambda_J \in K_J} \langle \nabla_u L(\hat{x}, \hat{u}, \hat{\lambda}), v\rangle \end{aligned}$$

Now suppose that ψ is continuously differentiable in (u, f) and consider directional derivatives of the upper level objective function ϕ. First we note the following result about directional derivatives of a composite function.

Lemma 3.2. (Demyanov and Rubinov, 1985) Let

$$\phi(u) = \psi(u, f^*(u))$$

and assume the following:

1. The function ψ : $R^m \times R \to R$ is continuously differentiable;

2. $f^* : R^m \to R$ possesses a directional derivative $f'^*(\hat{u}; s)$ at $\hat{u} \in R^m$ in every direction $s \in R^m$.

Then ϕ is directionally differentiable at $\hat{u}$ in every direction s, and

$$\phi'(\hat{u}; s) = \nabla_u \psi(\hat{u}, f^*(\hat{u})) \cdot s + \nabla_f \psi(\hat{u}, f^*(\hat{u})) f'^*(\hat{u}; s).$$

Now the problem of finding the steepest descent direction for the upper level optimization is defined as follows:

$$\text{(C)} \quad \begin{array}{ll} \text{minimize} & \phi'(\hat{u}; v) \\ \text{subject to} & \|v\| \leq 1 \end{array}$$

Here we assume that

$$\frac{\partial \psi(\hat{u}, f^*(\hat{u}))}{\partial f} =: R \geq 0 \tag{1}$$

Under this assumption, in view of (1) and Lemma 3.2, Problem (C) is

$$\min_{\|v\| \leq 1} [< \nabla_u \psi, v > + R \max_{\lambda_J \in K_J} < \nabla_u L, v >] = \min_{\|v\| \leq 1} \max_{\lambda_J \in K_J} < \nabla_u \psi + R \nabla_u L, v > \tag{2}$$

The set $\{v| \quad \|v\| \leq 1\}$ is compact and convex, and K_J is a compact convex set by Lemma 3.1. The objective function $< \nabla_u \psi + R \nabla_u L, v >$ is linear in v and affine in λ_J. Therefore, from the well-know minimax theorem, Problem (C) is equivalent to the problem

$$\text{(D)} \quad \max_{\lambda_J \in K_J} \min_{\|v\| \leq 1} < \nabla_u \psi + R \nabla_u L, v >$$

No we discuss Problem (D) in detail. First the minimization problem with respect to v in Problem (D), i.e. the problem

$$\min_{\|v\| \leq 1} < \nabla_u \psi + R \nabla_u L, v >$$

clearly has the minimum

$$-\|\nabla_u \psi + R \nabla_u L\|$$

This minimum is attained at

$$v = -\frac{\nabla_u \psi + R \nabla_u L}{\|\nabla_u \psi + R \nabla_u L\|}$$

in case of $\nabla_u \psi + R \nabla_u L \neq 0$ and at any v such that $\|v\| \leq 1$ in case of $\nabla_u \psi + R \nabla_u L = 0$. Therefore, noting that $\max(-(\ldots)) = -\min(\ldots)$, we have the equivalent form of Problem (D) as follows:

$$\text{(E)} \quad \min_{\lambda_J} \|\nabla_u \psi(\hat{u}, f^*(\hat{u})) + R[\nabla_u f^*(\hat{u}) + \sum_{j \in J(\hat{x}, \hat{u})} \lambda_j \nabla_u g_j(\hat{x}, \hat{u})]\|$$

$$\begin{array}{ll} \text{subject to} & G\lambda_J = -\nabla_x f(\hat{x}, \hat{u}) \\ & \lambda_j \geq 0 \qquad j \in J(\hat{x}, \hat{u}) \end{array}$$

Now we consider how to solve the the above problem (E). First we introduce the following notations:

$$\begin{bmatrix} C_1 \\ \vdots \\ C_m \end{bmatrix} := \nabla_u \psi(\hat{u}, f^*(\hat{u})) + R(\nabla_u f(\hat{x}, \hat{u}))$$

We assume that there exist k active inequality constraints (from the first to the k-th, for simplicity). Let $u \in R^m$ and

$$D := R^2 \begin{bmatrix} <\nabla_u g_1, \nabla_u g_1> & \dots & <\nabla_u g_1, \nabla_u g_k> \\ \vdots & & \vdots \\ <\nabla_u g_k, \nabla_u g_1> & \dots & <\nabla_u g_k, \nabla_u g_k> \end{bmatrix}, \qquad t := \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_k \end{bmatrix},$$

$$C := R \begin{bmatrix} \sum_{j=1}^{m} c_j \frac{\partial g_1}{\partial u_j} \\ \vdots \\ \sum_{j=1}^{m} c_j \frac{\partial g_k}{\partial u_j} \end{bmatrix}, \quad A := [\nabla_x g_1 \dots \nabla_x g_k], \quad b := -[\nabla_x f].$$

Then Problem (E) is rewritten as

$$\begin{aligned} \text{(F)} \quad & \min_t \sum_{j=1}^{m} c_i^2 + <t, Dt> + 2Ct \\ & \text{subject to} \quad At = b \quad t \geq 0 \end{aligned}$$

Since the Grammian matrix D is nonnegative definite, Problem (F) is a quadratic programming problem in t. Hence it can be solved rather easily by certain quadratic programming technique and we can obtain λ_J. The search direction v in the upper level optimization can be computed via (2) by using the value of λ_J.

4 An Optimization Method for Two–Level Problems

Now we provide an optimization method for two–level Problem (T) based on the results given in the preceding sections. We assume the following in Problem (T):

1. The upper level objective function ψ is continuously differentiable and the lower level function f, and the constraint functions g_j are twice continuously differentiable.

2. The set of optimal solutions is nonempty and compact for every u.

3. The Cottle constraint qualification holds at every solution.

4. For each u, the optimal solution is unique.

5. Either the other assumptions in Theorem 2.1 are satisfied (Case 1), or the assumptions in Theorem 2.3 are satisfied (Case 2).

6. $R \geq 0$ holds.

Then we have the following optimization algorithm for solving Problem (T):
An optimization algorithm for Problem (T):

Step 1: Take an initial point u^1 and a parameter for convergence $\varepsilon > 0$. Set $k = 1$ and $\Pi = 0$.

Step 2: Solve the lower problem for $u = u^k$. In Case 1, go to subproblem (1) described below. In Case 2, go to subproblem (2).

Step 3: If $\Pi = 1$, the convergence condition is supposed to be satisfied and therefore adopt u^k and x^k as optimal solutions in the upper and lower levels, respectively, and stop. Otherwise put $u^{k+1} = u^k + \alpha^k v^k$ and conduct the line search to decide an optimal α^k.

Step 4: Let $k = k + 1$ and return to Step 2.

Subproblem (1) for finding a descent direction in the upper level

Step 11: Compute $s^k = \nabla_v \psi(u^k) + \nabla_f \psi \cdot \nabla_u L(\hat{x}, u^k)$

Step 12: If $\|s^k\| < \varepsilon$, put $\Pi = 1$ and return to Step 3. Otherwise get

$$v = -\frac{s^k}{\|s^k\|}$$

and return to Step 3.

Subproblem (2) for finding a descent direction in the upper level

Step 21: Formulate Problem (F) with $u^k, \hat{x}$.

Step 22: Solve Problem (F) as a quadratic programming problem. Denote the solution by λ^k.

Step 23: By using $u^k, \hat{x}, \lambda^k$ and so on, compute

$$\nabla_u \psi + R \nabla_u L.$$

If $\|\nabla_u \psi + R \nabla_u L\| < \varepsilon$, then let $\Pi = 1$ and return to Step 3. Otherwise, compute

$$v^k = -\frac{\nabla_u \psi + R \nabla_u L}{\|\nabla_u \psi + R \nabla_u L\|}$$

and return to Step 3.

5 An application to two-level multiobjective optimization problems

In this section we consider the following multiobjective optimization problem as the lower level problem:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f(x,u) = (f_1(x,u),\ldots,f_p(x,u)) \\ \text{subject to} & g_j(x,u) \leq 0, \qquad j = 1,\ldots,q \end{array}$$

We assume that this problem is scalarized by Tchebycheff norm minimization with the origin as the reference point. Namely we consider the following problem:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \max_i f_i(x,u) \\ \text{subject to} & g_j(x,u) \leq 0, \qquad j = 1,\ldots,q \end{array}$$

This problem is written in the following equivalent form:

$$\begin{array}{lll} \underset{x,z}{\text{minimize}} & z & \\ \text{subject to} & f_i(x,u) - z \leq 0, & i = 1,\ldots,p \\ & g_j(x,u) \leq 0, & j = 1,\ldots,q \end{array}$$

And we suppose that the upper level problem is given by

$$\underset{u}{\text{minimize}} \qquad \phi(u) = \psi(u, z^*(u))$$

where $z^*(u)$ is the optimal value of the above lower level problem.

In this case the linear independence constraint qualification is written as

$$\left\{ \begin{bmatrix} \nabla_x & f_i \\ & -1 \end{bmatrix} (i \in I), \begin{bmatrix} \nabla_x & g_j \\ & 0 \end{bmatrix} (j \in J) \right\} \text{ are linearly independent.}$$

This condition is weaker than the condition that $\{\nabla_x f_i(i \in I), \nabla_x g_j(j \in J)\}$ are linearly independent and stronger that the condition that $\{\nabla_x g_j(j \in J)\}$ are linearly independent.

On the other hand, the Cottle constraint qualification in this case is written as follows: there exist $h \in R^n$ and $\alpha \in R$ such that

$$\begin{array}{ll} \langle \nabla_x f_i(\hat{x},\hat{u}), h\rangle < \alpha & \forall i \in I(\hat{x},\hat{u}), \\ \langle \nabla_x g_j(\hat{x},\hat{u}), h\rangle < 0 & \forall j \in J(\hat{x},\hat{u}). \end{array}$$

However this condition is equivalent to the condition that there exists $h \in R^n$ such that

$$\langle \nabla_x g_j(\hat{x},\hat{u}), h\rangle < 0 \qquad j \in J(\hat{x},\hat{u}).$$

6 Conclusion

In this paper we have proposed an optimization method for two-level optimization problems. In the method, the uniqueness of the optimal solution and the Lagrange multiplier vector is taken into account. It is applicable to a large class of problems. An application to two-level multiobjective optimization has been also dealt with.

References

Demyanov, V. F. and Rubinov, A. M. (1985). Quasidifferential Calculus, IIASA, Laxenburg.

Fiacco, A. V. (1976). Sensitivity Analysis for Nonlinear Programming Using Penalty Methods, *Math. Prog.,* Vol. 10, pp. 287–311.

Fiacco, A. V. (1983). Introduction to Sensitivity and Stability Analysis in Nonlinear Programming, Academic Press, New York.

Gauvin, J. and Dubeau, F. (1982). Differential Properties of the Marginal Function in Mathematical Programming, *Math. Prog. Study,* Vol. 19, pp. 101–119.

Gauvin, J. and Tolle, J. W. (1977). Directional Stability in Nonlinear Programming, *SIAM J. Contr. and Optim.,* Vol. 15, pp. 294–311.

Mesarovic, M. D., Macko, D. and Takahara, Y. (1970). Theory of Hierarchical Multilevel Systems, Academic Press, New York.

Mifflin, R. (1977). Algorithm for Constrained Optimization with Semismooth Functions, *Math. of Oper. Res*, Vol. 2, pp. 191–207.

Rockafellar, R. T. (1984). Directional Differentiability of the Optimal Value Function in a Nonlinear Programming Problem, *Math. Prog. Study,* Vol. 21, pp. 213–226.

Shimizu, K. and Ishizuka, Y. (1985). Optimality Conditions and Algorithms for Parameter Design Problems with Two-Level Structure, *IEEE. Trans. Auto. Contr.*, Vol. AC-30 pp. 986–993.

Tanino, T. and Ogawa, T. (1984). An Algorithm for Solving Two-Level Convex Optimization Problems, *Int. J. Systems Sci.,* Vol. 15, pp.163–174.

Non-Objective-Submerged and Interactive Approach to Multiobjective Linear Programming

Yanzhang Wang and Zhongtuo Wang
Institute of Systems Engineering
Dalian University of Technology
116024 Dalian, P. R. of China

Abstract

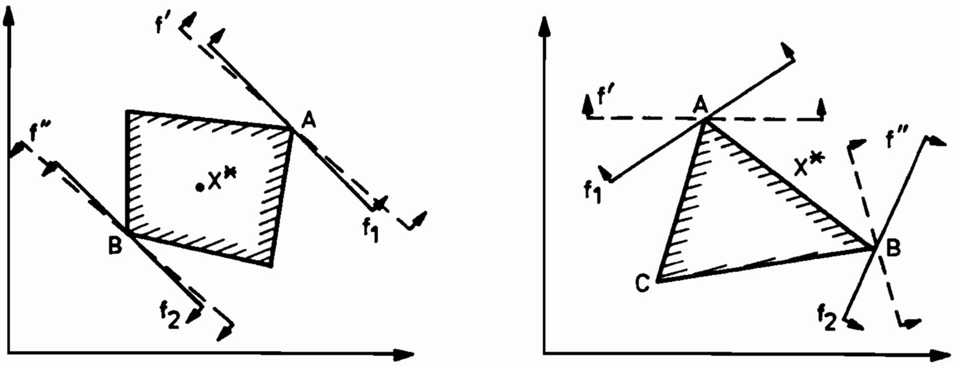
In this paper, objective submerged problems and limitation existing in the methods currently used for Multi-Objective Linear Programming (MOLP) are analysed related to real-world processes. The concepts of consistency measure matrix, basical reference solution set and non-objective-submerging are presented on the basis of convex set theory, and a fast algorithm of solving the matrix is given. Furthermore, non-objective-submerged searching model is built according to the principle of equal satisfying integration of objectives. Finally, the non-objective-submerged and interactive approach is implemented, where non-objective-submerging is existing and exactly reflecting the aspiration or the preferences of decision makers.

I. Introduction

In real-world decision processes, Multiple Objective Linear Programming (MOLP) is still one of the useful methods, because linear models are convenient to build and recognized by the experience of decision makers. Up-to-now many methods for solving MOLP problems (Chankong and Haimes, 1983; Zeleny, 1982; Lee, 1972; Chunjun Zhao, 1987) have been proposed, and they are effective for some situations of the decision making. But, as the problems of decision making becoming complicated and enlarging in scales, especially in the cases where many objectives are incomparable in sense or some of them are conflicting, these methods can not satisfy the demand of the decision making.

These methods depend mainly on the utility theory or/and the reference point, for example, weighting coefficients or goal programming method. In fact, the feature of them is to find a criterion so that it can be used to coordinate the profits of objectives. The criterion is usually a utility function or a distance to the reference point in decision space, and it is a linear scalarizing function for the convenience of applying the effective LP method to solve it. However, in general cases the subjective value measures or preferences of decision makers are not linear relation with the objectives. Thus, linear criterion can not effectively reflect the satisfying integration of objectives, that is the limitation of above methods in application.

For a linear criterion, the substitution of every objective is constant. This means that one objective can be constantly substituted by another and the marginal substitution is not diminishing. This situation is called as objective submerging. This can be illustrated in two cases (a) and (b) in Figure 1.1. Linear criterion f is either f' or f'', and effective solution point is either A or B. From Figure (b) to see, if the slope of f is between the slope of line AB and objective function f_1, then objective f_1 submerges objective f_2. In the same way, if the slope of f is between the slope of BA and f_2, then f_2 submerges f_1. As we know by LP theory, in the condition of linear criterion, effective solution point must be on the vertex of the feasible solution set, but it is usually dissatisfied solution. Generally a satisfying solution is possibly the point X^* in Figure 1.1. This shows the effective solution point may not be the vertex. This is why the reference point method is better than other, in which the objective submergence is limited in a small scope and the substitution of objectives stops at its reference point although constantly substituted.



(a) The case of f_1 conflicting with f_2

(b) The case of f_1 not conflicting and consistent with f_2

Figure 1.1 Effective solution situations of a linear criterion

In addition, interactive approaches are now widely used in real-world decision processes because for the complicated decision making problems the experiential information of decision makers is needed. However, an effective approach has a reasonable division of work between decision makers (or human) and models (or computer). For the MOLP problems, if a linear criterion function is defined, it will be not adaptable to make the interaction, because the substitution of objectives given by the linear function is not always satisfactory and often confuses the determination of the preferences of decision makers.

For these reasons, in this paper, a Non-Objective-Submerged and Interactive Approach (NOSIA) to MOLP has been presented. Where, integration or substitution of objectives is only determined on the basis of the preferences, and it gives a good situation to facilitate the generation of the preference information.

II. Basical mathematics concepts of NOSIA

Suppose the MOLP problem is as following

$$\begin{aligned} \max \quad & Z = CX \\ \text{s.t.} \quad & AX = b, \quad X \geq 0 \end{aligned} \qquad \text{(MOLP)}$$

where, $X \in R^n$, $C \in R^{p\times n}$, $A \in R^{m\times n}$, $Z \in R^p$, $b \in R^m$; n is the number of variables, m the number of constraints, and p the number of objectives. For convenience, assume $I = \{1,2,\ldots,m\}$, $J = \{1,2,\ldots,n\}$, $K = \{1,2,\ldots,p\}$. Moreover, for $\forall\ i \in K$, C_i denotes the row vector of matrix C, and for $\forall\ i \in I$, a_i the row of A. Meanwhile, assume the feasible solution set of MOLP is $\mathbf{X}$, and

$$\mathbf{X} = \{\, X \mid AX = b,\ X \geq 0 \,\}$$

Definition 1. Decision support matrix is denoted by D, and

$$D \triangleq [\, d_{ij} \,]$$

here

$$d_{ij} \triangleq C_i X_j^* \qquad \forall\ i,j \in K$$

and X_j^* subjects to $C_j X_j^* = \max\limits_{X\in\mathbf{X}} C_j X$.

Easy to see, d_{ii} is the optimal value of the i-th objective when it is optimized as a single objective. Let $q^* = [\, d_{11}\, d_{22}\, \ldots\, d_{pp} \,]^t$, and obviously it is the ideal solution point of MOLP problem, but in general cases it is impossible to arrive. If it could be arrived, it is the optimal solution point of MOLP problem. Corresponding to this, assume $q^- = [\, \min\limits_{j\in K} d_{1j}\ \min\limits_{j\in K} d_{2j}\ \ldots\ \min\limits_{j\in K} d_{pj} \,]^t$, and it is called as the valley point of MOLP problem.

Furthermore, define respectively the following objective state set $\mathbf{Q}$ and vector set of preference weight coefficient $\mathbf{W}$.

$$\mathbf{Q} \triangleq \{\, q \mid q \in R^p,\ q^- \leq q \leq q^* \,\}$$

here q is an objective state vector, q^* and q^- subject to the definition 1.

$$\mathbf{W} \triangleq \left\{ W \mid W \in R^p,\ \text{for } \forall\ i \in K,\ \text{the element of } W,\ W_i \geq 0 \text{ and } \sum_{i\in K} W_i = 1 \right\}$$

Theorem 1. Suppose $W \in \mathbf{W}$, then for the trade-off objective state vector $q = DW$ and $q \in \mathbf{Q}$, there exists a X such that $CX = q$ and $X \in \mathbf{X}$.

PROOF: From definition 1 we have:

$$\begin{aligned} q &= DW = [\, d_{ij} \,]\, W \\ &= C\, [\, X_1^*\, X_2^*\, \ldots\, X_p^* \,]\, W \end{aligned}$$

Taking $X = [X_1^* \, X_2^* \, \ldots \, X_p^*] \, W$, and from definition 1, for $\forall \, i \in K, \;\; X_i \in \mathbf{X}$. Moreover, by the definition of the set $\mathbf{W}$, for $\forall \, i \in K, \;\; W_i \geq 0, \;\; \sum_{i \in K} W_i = 1$, and by the LP theory $\mathbf{X}$ is convex. Hence we have $X = W_1 X_1^* + W_2 X_2^* + \cdots + W_p X_p^* \in \mathbf{X}$.

□

Definition 2. Basical reference solution set $\mathbf{X}_b$ is defined as

$$\mathbf{X}_b \triangleq \{ X \mid X = [X_1^* \, X_2^* \, \ldots \, X_p^*] \, W, \; W \in \mathbf{W} \}.$$

Basical reference solution set $\mathbf{X}_b$ and objective state vector $q = DW$ can give decision makers a reference basis to determine their preference values.

Definition 3. Noninferior solution set of MOLP $\mathbf{X}^*$ is defined as follows

$$\mathbf{X}^* \triangleq \{ X^* \mid X^* \in \mathbf{X}, \text{ and there exists no other feasible } X \text{ (i.e. } X \in \mathbf{X}) \text{ such that } C_j X \geq C_j X^* \text{ for } \forall \, j \in K \text{ with strict inequality for at least one } j \, \}.$$

Theorem 2. In the condition of definition 1, if for $\forall \, i \in K, \;\; X_i^*$ is a unique maximum point of objective $C_i X$, then $X_i^* \in \mathbf{X}^*$.

PROOF: Assume $X_i^* \overline{\in} \mathbf{X}^*$, then there exists at least one $X' \in \mathbf{X}^*$ such that $CX' \geq CX_i^*$ meaning that $C_i X' \geq C_i X_i^*$. This is conflicting with the assumption that X_i^* is a unique maximum point of $C_i X$.

□

Definition 4. Assume that the consistency of objectives Z_i and Z_j, for $\forall \, i, j \in K$, is measured by η_{ij} and

$$\eta_{ij} = (V_{ij} V_{ij}^t)^{1/2}$$

here

$$V_{ij} = \frac{C_i}{(C_i C_i^t)^{1/2}} + \frac{C_j}{(C_j C_j^t)^{1/2}}$$

Then the consistency measure matrix η is defined as

$$\eta \triangleq [\, \eta_{ij} \,]$$

Obviously, we have $0 \leq \eta_{ij} \leq 2$ for $\forall \, i, j \in K$. If $\eta_{ij} = 2$, objective Z_i is completely consistent with objective Z_j and vector C_i linear dependence with vector C_j meaning that one of them can be substituted by another. If $\eta_{ij} = 0$, objective Z_i is conflicting with objective Z_j and in the same way vector C_i is linear dependent with vector C_j. Meanwhile, we can see that the consistency measure matrix η is symmetrical, i.e. $\eta_{ij} = \eta_{ji}$, and $\eta_{ij} = 2$ for $\forall \, i \in K$.

III. Non-objective-submerged and interactive approach (NOSIA)

Basical idea of NOSIA is a reasonable division of work between decision makers and models in the decision processes. As mentioned in section I, using a linear criterion to coordinate the profits of objectives, the unreasonable substitution of the objective submerging may appear. This means that the methods have the function to decide the substitution of objectives on themselves. However, this often can not correctly reflect the aspiration of decision makers. In NOSIA the authority to determine the substitution is fully returned to decision makers, and the models only have the function to supply integrated information to decision makers and to find out a noninferior or "best" solution on the preference of decision makers. The main feature of the NOSIA is that the reasonable relation of objective set and preference has been built, which can give the exact reflection of the preference information from decision makers. NOSIA can be separated as the following two stages:

1. Find the consistency measure matrix η and decision support matrix D

The aim of the stage is to give a good situation to decision makers facilitating the generation of the preference information through computing the matrices η and D. From definition 4, to calculate the matrix η is easy and it only needs to compute $\frac{1}{2}p(p-1)$ values of η_{ij}. The direct method to compute the matrix D is optimizing solely every single objective, but it is stupid. Here a fast algorithm of solving the matrix D is given by applying the information of the matrix η, and the basic steps of the algorithm are as follows:

Step 1. For $\forall\, i \in K$, put Z_i in order such that they are constructing a longest path where the consistency measure η_{ij} is considered as a step length.

Step 2. According to the consistency order of objectives given in step 1, first solve a feasible solution and the optimal solution of the objective Z_1 at the start of the path. Then, employing the basis B obtained in the optimization of the preceding objective, continue the simplex iteration to optimize the present objective, and to this one by one along the path until the matrix D is found.

This procedure can be illustrated as the Figure 3.1, where objectives Z_1, Z_2, and Z_3 are arranged in consistency order and the simplex iteration makes in the direction of arrows. Clearly, after the optimal solution of objective Z_1 is solved, it only needs two steps of the simplex iteration to get the optimal solutions of objectives Z_2 and Z_3. Generally, the time of computing matrix η and arranging the order of objectives is proportional to $O(p^2)$, the time of one step simplex iteration is proportional to $O(m \times n)$, and usually we have $m, n \gg p$, therefore this algorithm is faster.

2. Procedure of non-objective submerged searching and interaction with decision maker

This procedure consists of two steps of acquiring the preference information from decision maker and searching noninferior solution by non-objective-submerged searching model.

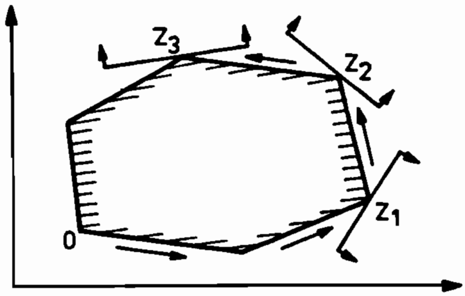


Figure 3.1 A procedure of solving matrix D

Step 1. Integrating the information such as decision support matrix D, basical reference solution set $\mathbf{X}_b$, consistency measure matrix η and subjective preferences, decision maker can determine the reference point q^r or preference weighting coefficient W.

If decision maker can not clearly give the preference, we can use the information given by objective states q^* and q^- to construct a satisfied degree vector $\mathbf{u}$ such that $\mathbf{u} \in [0,1]^p$ and for $\forall\, i \in K$, u_i, the element of $\mathbf{u}$, is defined as

$$u_i \triangleq \frac{q_i - q_i^-}{q_i^* - q_i^-} \tag{3.1}$$

here q_i is an element of vector q and $q \in \mathbf{Q}$.

Step 2. Noninferior solution is searched by the non-objective-submerged searching model (NOSSM). For the convenience of understanding, we first introduce the following basical NOSSM.

$$\begin{aligned}
\max \quad & Z = \sigma \\
\text{s.t.} \quad & AX = b \\
& CX - q^r\sigma \geq 0 \\
& X \geq 0, \quad \sigma \geq 0.
\end{aligned}$$

Here the exact meaning of variable σ is the variable coordinating the profits of objectives on the basis of the reference point q^r, which enables to coordinate them on an equal importance, therefore no substitution of objectives or no objective submerging is existing. In other words, the NOSSM can give the equal rate of improving to every objective, and decision maker can adjust the relation of objective profits by changing the value of the reference point q^r.

But, the solution given by above model sometime is not a noninferior solution of MOLP. As the Figure 3.2 shows, where objectives Z_1 and Z_2 are conflicting each other, i.e. $\eta_{12} = 0$. If we take the reference point as q^r in the Figure, evidently, the solution of above model is $\sigma = 1$, this means that all of the objectives can not be improved on the equal rate. But keeping the values of objectives Z_1 and Z_2 unchanged, we can improve the value of objective Z_3 and achieve the noninferior solution point X^*.

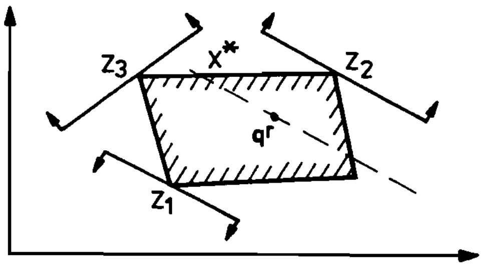


Figure 3.2 A case of the solution given by basic NOSSM

For the real MOLP problems, it is generally permitted to improve solely one single objective if it does not effect the profits of other objectives. So we revise the basic NOSSM as

$$\begin{aligned} \max \quad & Z = M\sigma + \sum_{i \in K} \rho_i \\ \text{s.t.} \quad & AX = b \\ & CX - q^r\sigma - \rho = 0 \\ & X \geq 0, \quad \sigma \geq 0, \quad \rho \geq 0. \end{aligned}$$

where $\rho \in R^p$, ρ_i is an element of ρ for $\forall\, i \in K$, and M is a number large enough to enable the $M\sigma$ dominating the term $\sum_{i \in K} \rho_i$. For convenience this model is called as NOSSM.

Theorem 3. If $[\,X^{ot}\,\sigma^o\,\rho_1^o\,\rho_2^o\,\ldots\,\rho_p^o\,]^t$ is the optimal solution of the NOSSM, then X^o is a noninferior solution of original MOLP problem, i.e. $X^o \in \mathbf{X}^*$.

PROOF: Since $[\,X^{ot}\,\sigma^o\,\rho_1^o\,\rho_2^o\,\ldots\,\rho_p^o\,]^t$ is the optimal solution of the NOSSM, we have $AX^o = b$, $X^o \geq 0$, i.e. $X^o \in \mathbf{X}$. Assume $X^o \overline{\in} \mathbf{X}^*$, then there exists a $X \in \mathbf{X}$ such that $CX \geq CX^o$ and existing at least one $i \in K$ such that $C_iX > C_iX^o$. Therefore we have $\rho_i' > \rho_i^o$ and $[\,X^t\,\sigma^o\,\rho_1^o\,\ldots\,\rho_i'\,\ldots\,\rho_p^o\,]^t$ is a feasible solution of the NOSSM, and

$$M\sigma^o + \sum_{i \in K\setminus\{i\}} \rho_j^o + \rho_i' > M\sigma^o + \sum_{i \in K\setminus\{i\}} \rho_j^o + \rho_i^o.$$

This is conflicting with the assumption that $[\,X^{ot}\,\sigma^o\,\rho_1^o\,\rho_2^o\,\ldots\,\rho_p^o\,]^t$ is the optimal solution of NOSSM, i.e. $X^o \in \mathbf{X}^*$. □

From the NOSSM an effective interaction can be easily made, and the determination of the q^r is convenient too. Generally, there are three ways to define the q^r.

(1) Take the reference point as the q^r;

(2) If given W, we can take $q^r = DW$;

(3) Applying the concept of the satisfied degree, and from the formula (3.1), we have

$$q_i = q_i^- + (q_i^* - q_i^-)u_i$$

Let $u_1 = u_2 = \cdots = u_p = \sigma$, and substitute $q^r\sigma$ in the NOSSM by q. Thus the NOSSM is changed into the equal satisfied degree searching model.

IV. An example

Given the MOLP as

$$\begin{aligned} \max \quad & Z_1 = 2x_i + 5x_2 \\ & Z_2 = 4x_1 + x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 10 \\ & x_1 \leq 8 \\ & x_2 \leq 6 \\ & x_1\ , \ x_2 \geq 0 \end{aligned}$$

the procedure of solving this problem by the NOSIA is as follows:

Step 1. The decision support matrix D, the consistency measure matrix η, and two optimal solutions X_1^* and X_2^* corresponding to the two objectives are respectively computed as

$$D = \begin{bmatrix} 38 & 26 \\ 22 & 34 \end{bmatrix} \qquad \eta = \begin{bmatrix} 2 & 1.72 \\ 1.72 & 2 \end{bmatrix}$$

$$X_1^* = \begin{bmatrix} 4 \\ 6 \end{bmatrix} \qquad X_2^* = \begin{bmatrix} 8 \\ 2 \end{bmatrix}$$

Then

$$\mathbf{X}_b = \left\{ X \mid X = \begin{bmatrix} 4 \\ 6 \end{bmatrix} W_1 + \begin{bmatrix} 8 \\ 2 \end{bmatrix} W_2,\ W_1, W_2 \geq 0,\ W_1 + W_2 = 1 \right\}$$

Taking

$$W = [0.5\ 0.5]^t$$

we have

$$q^r = \begin{bmatrix} 38 \\ 22 \end{bmatrix} \times 0.5 + \begin{bmatrix} 26 \\ 34 \end{bmatrix} \times 0.5 = \begin{bmatrix} 32 \\ 28 \end{bmatrix}$$

Step 2. Let $M = 60$, the NOSSM is constructed as

$$
\begin{array}{lllll}
\max & Z = 60\sigma + \rho_1 + \rho_2 & & & \\
\text{s.t.} & x_1 + x_2 & & \leq & 10 \\
 & x_1 & & \leq & 8 \\
 & x_2 & & \leq & 6 \\
 & 2x_1 + 5x_2 - 32\sigma - \rho_1 & & = & 0 \\
 & 4x_1 + x_2 - 28\sigma - \rho_2 & & = & 0 \\
 & x_1 , x_2 , \sigma , \rho_1 , \rho_2 & & \geq & 0
\end{array}
$$

The optimal solution of the model is $[6\ 4\ 1\ 0\ 0]^t$, i.e.

$$X^* = \begin{bmatrix} 6 \\ 4 \end{bmatrix}, \qquad Z = \begin{bmatrix} 32 \\ 28 \end{bmatrix}, \qquad \sigma = 1$$

If revising the weighting vector such as $W = [0.3\ 0.7]$, we have

$$X^* = \begin{bmatrix} 6.8 \\ 3.2 \end{bmatrix}, \qquad Z = \begin{bmatrix} 29.6 \\ 32.5 \end{bmatrix}$$

Easy to see, the reflection of the objectives from revising the preference information is good and clear.

V. Conclusion

Through above analysis and application to real decision making problem, the NOSIA is proved to be an effective method for solving MOLP problems, especially for the case where some objectives are conflicting each other, or where the equilibrium of the profits for multiobjective is needed. The algorithm of solving decision support matrix given in this paper not only decreases the time of computing, but it also can give the information about consistency measure of objectives. The NOSSM can avoid the problem of objective submerging and exactly reflect the subjective value concepts or the preferences of decision makers. Consequently the method facilitates the interaction of decision makers with the model, and it shortens the time taken by the process of decision making.

References

Chankong, V. and Haimes, Y.Y. (1983). Multiobjective Decision Making. North-Holland, New York.

Chunjun Zhao, (1987). DIDASS–Dynamic Interactive Decision Analysis and Support System. *Systems Engineering*, No. 4.

Lee, S.M. (1972). Goal Programming for Decision Analysis. Auerbach Pub. Inc., Philadelphia.

Lewandowski, A. (1982). A Program Package for Linear Multiple Criteria Reference Point Optimization Short User Manual. WP-82-80, IIASA.

Rockafellar, R.T. (1972). Convex Analysis. Princeton University Press.

Wierzbicki, A.P. (1979). A Methodological Guide to Multiobjective Optimization. WP-79-122, IIASA.

Zeleny, M. (1982). Multiple Criteria Decision Making. McGraw-Hill, New York.

Characterization of Efficient Decisions

Petra Weidner
Martin-Luther-University Halle-Wittenberg
Halle, Germany

1 Introduction

Nowadays the publications in vector optimization are covering a wide field. Procedures for practical applications at one hand and general choice theory at the other one seem to be two extreme points the problems and results of which ones do not influence each other. But, indeed, they should. What we are going to offer in this article is a general approach to decision problems which is special enough to deliver new statements even for Pareto-optimality and in this way to answer questions arising in modelling multicriterial problems and in applying solution methods to them.

The decision problem we consider can be formulated in the following form:

- A set F of possible decisions is given, and the analyst has to help the decision maker in finding the best decision with respect to his preference relation.

Let us assume that F is a subset of a linear space Y and that the preferences can be aggregated by means of a set $D \subset Y$ such that

- y^1 is preferred to y^2 by the decision maker iff $y^2 \in y^1 + (D \setminus \{\emptyset\})$. D is called the domination set of the problem. It has not necessarily to be a cone.

The final decision belongs to the set

$$\mathrm{Eff}\,(F,D) := \{\bar{y} \in F \mid \nexists\, y \in F :\ \bar{y} + (D \setminus \{\emptyset\})\}$$

of efficient elements of F with respect to D.

In most cases F is the image of a control set X via an objective mapping $f : X \to Y$. Then the original $f^{-1}(\mathrm{Eff}\,(F,D))$ will be denoted by $\mathrm{Eff}_f(X,D)$ and its elements will be referred to as efficient points of X with respect to f and D.

It is the aim of our paper to point out the advantage of this general approach and to indicate the scope of research done on this basis. Detailed results as well as references to literature have to be omitted because of page limitation, but can be found in (Gerth and Weidner, to appear) and in (Weidner, 1983a, 1983b, 1985a, 1985b, 1986a, 1986b, 1987a, 1987b, 1988a, 1988b, 1988c, to appear, in preparation).

If in the following Y has to be equipped with a topology, then it will be assumed to be a linear topological space.

2 Advantages of the use of arbitrary domination sets in linear spaces

(a) For Y being the k-dimensional Euclidean space R^k and D being the non-negative orthant R^k_+ of this space, $\mathrm{Eff}_f(X, R^k_+)$ turns out to be the set of Pareto-minima of X with respect to f. Thus, consequences for Pareto-minimality from results in the general framework become obvious. Moreover, the general assumptions result in clearer proofs, in new insights in the necessity of conditions and in the source of certain properties.

(b) At the beginning of the decision process the efficient point set will usually consist of more than one point, though only one element has to be chosen. The efficient point set can become smaller, if the domination set is extended. Wierzbicki, for example, defined in a normal space Y and for a closed convex cone D D_ε-minimal elements (Wierzbicki, 1980), which are just efficient points with respect to $D_\varepsilon \setminus -D_\varepsilon$, where D_ε is the so-called ε-coneneighbourhood $D_\varepsilon := \{y \in Y \mid \min_{d \in D} \|y - d\| < \varepsilon \|y\|\}$ of D. The investigation of efficiency with respect to convex cones goes back to Hurwicz (1958). Yu (1974) studied cone-efficiency in R^k in connection with domination structures. This concept was extended to efficiency with respect to convex sets by Bergstresser, Charnes and Yu (1976). Chew (1984) considered efficiency in linear spaces with respect to additive semigroups D.

(c) When modelling the decision problem, different uncertainties can exist because of data perturbations, doubts of the decision maker or objectives which can not be expressed as functions. Then the formulation of preferences between points which do not differ very much becomes difficult, and it is sensible to choose a set D that does not intersect some neighborhood of the origin O. Such a set D could be bounded by an hyperbola or in $Y = R^k$ be a set $\varepsilon + R^k_+$ with $\varepsilon \in R^k_+$ as it is introduced by Loridan (1984).

(d) Y can stand for a space of trajectories or functions. For example, the choice of control variables for a problem that also depends on random variables can be modelled:

- Let X be the set of control variables, T the set of random variables and $g : X \times T \to R$ be the function the value of which is to be maximized. Thus, we have the problem

$$\begin{aligned} &g(x,t) \to \max \quad \forall\, t \in T, \\ &\text{s.t.} \quad x \in X. \end{aligned}$$

x^1 is preferred to x^2 iff $g(x^1, t) \geq g(x^2, t) \quad \forall\, t \in T$.

Consequently, the preference relation in the space Y of all mappings from T into R is generated by the domination set

$$D = \{g_x :\; T \to R \mid g_x(t) \leq 0 \quad \forall\, t \in T\}.$$

(e) The decision maker's preferences are commonly given on the set $f(X)$, though the set X is the set of control variables. If the preferences are transferred to X via f^{-1}, then the domination set in the space that contains X will have less convenient properties than in F in general, but our concept may deliver statements about the efficient controls.

(f) Results from literature presented under different assumptions can be compared, among them those mentioned in (b) and (c), but also statements of Rander (1967), who studied the efficient point set of a convex set F in the space of all bounded sequences of real numbers with respect to the non-negative orthant, and of Peleg (1970), who applied Rander's results to prices for optimal consumption plans.

3 State of the art

Investigating decision problems of the type defined above, we have given conditions for

1. the existence of efficient points (Weidner, 1983b, 1985b, 1988c),
2. the efficiency of all feasible points (Weidner, 1988a, to appear),
3. the non-efficiency of interior points of the feasible set in the decision space and in the control space (Weidner, 1985a, 1985b, 1988c, to appear),
4. the global efficiency of locally efficient points (Weidner, 1985a, 1985b, 1988c),
5. extensions, restrictions and decompositions of the feasible point set which extend, restrict or do not alter the optimal point set (Weidner, 1983b, 1985b, 1986b, 1987a, 1988b, 1988c),
6. the determination of efficient points, if further preferences are added during the solution process, if domination sets are decomposed, united or changed otherwise (Weidner, 1985b, 1987a),
7. interdependencies between domination set and objective function (Weidner, 1985b, 1987b, to appear),
8. the information about efficient points by means of domination sets defined immediately in the control space (Weidner, 1987b, to appear),
9. the characterization of efficient points as optima of scalar-valued linear and nonlinear functions (Gerth and Weidner, to appear, Weidner 1985b, 1986a, 1988b, 1988c),
10. the efficiency of infimal points (Weidner, 1985b),
11. duality (Weidner, 1983a).

The results obtained give answers to the following questions which are essential in applying most of the vector optimization procedures:

- Can the feasible point set be made convex and/or closed without influencing the optimal point set? (Weidner, 1986b)
- Is it sufficient to restrict one's attention to the previously optimal points, if further preferences become obvious? (Weidner, 1985b, 1987a)
- Can the efficient points be determined by looking for optimality in parts of the feasible point set (Weidner, 1985b, 1988c)

- Does the efficient points set with respect to the domination set D coincide with that with respect to the closed convex cone generated by D? (Weidner, 1988b)
- Can efficiency with respect to polyhedral cones in Euclidean spaces always be transformed into Pareto-optimality? (Weidner, 1985b, 1987b, to appear)
- Do there exist equivalences between the Pareto-optima of vector optimization problems with linear objective functions and the efficient points with respect to polyhedral cones in the control space? (Weidner, 1987b, to appear)
- What changes in the objective functions extend, restrict or do not alter the set of Pareto-optima? Can objective functions f_i which are linear combinations of other objective functions be omitted? (Weidner, 1988a, 1988c, to appear)

None of the answers is trivial, and in each case the general approach proved to be helpful in finding necessary assumptions and counterexamples to statements asserted to be self-understood in some publications.

Basic requirements for several desirable properties of the efficient point set are the so-called domination property $F \setminus \text{Eff}\,(F, D) \subset \text{Eff}\,(F, D) + D$ and the condition $D + D \subset D$, which is fulfilled by a much broader class of sets than that of the convex cones. A discussion of these assumptions one can find in (Weidner, 1983b, 1985a, 1985b, 1988b) and (Weidner, 1985b, 1986b), respectively.

Various authors studied optimality in vector optimization as efficiency with respect to a set $D \setminus -D$, where D is a convex cone. In (Weidner,1985b) the topics (1), (3)-(6) and (9) as well as the domination property are investigated for this efficient point set with D being an arbitrary set. Moreover, efficiency with respect to D is compared with that with respect to $D \setminus -D$.

Different algorithms for the solution of multicriterial optimization problems determine the weakly efficient point set instead of the efficient point set, since the weakly efficient point set contains the efficient point set and often has more suitable properties. In our terminology, weakly efficient elements of F with respect to D are efficient elements of F with respect to the topological interior of D. We have dealt with the items (1), (3)-(6), (9) and (10) for weakly efficient points in (Weidner, 1985b), where, besides, weakly efficient points are characterized by means of the tangent cone, and relations between the weakly efficient point set and the efficient point set are stated. Further scalarization results for weakly efficient points one can read in (Gerth and Weidner, to appear, Weidner, 1986a, 1988b, 1988c). Closeness and compactness of the weakly efficient point set are observed in (Weidner, 1985b, 1988b).

A counterpart to the weakly efficient points are the properly efficient points, which form a mathematically well-behaved subset of the efficient point set. In (Weidner, 1983a, 1983b, 1985a, 1986a, 1988b) different sets of properly efficient points defined by other authors are compared and adapted to arbitrary domination sets.

A function $z : Y \to R$ is said to be strictly D-monotone iff $y^1 \in y^2 + (D \setminus \{\emptyset\})$ implies $zy^1 > zy^2$. Since each point of F in which some strictly D-monotone functional attains its minimum on F is an efficient element of F with respect to D , every set of strictly D-monotone functionals can be used in order to define proper efficiency. Another approach to proper efficiency is offered in the following definition:

- An element y of F is called a properly efficient point of F with respect to D and to the family Z of sets $H \supset D \setminus \{\emptyset\}$ with specified properties iff $\exists H \in Z$: $y \in \mathrm{Eff}\,(F, H)$.

In (Gerth and Weidner, to appear) we prove that the optima of strictly D-monotone functionals are properly efficient points according to the last definition and vice versa. Some of these statements and the underlying separation theorems are published in (Elster and Gõpfert, 1987) where all results except those concerning Lagrange type duality are picked from our manuscript for (Gerth and Weidner, to appear).

A discussion of the optimal point sets mentioned in the framework of domination structures is presented in (Weidner, 1985a). In (Weidner, 1985b, 1986a, and in preparation) we underline the close connection between efficiency and optimality with respect to binary relations in detail, thereby also taking into consideration classical results e.g. from (Neumann, 1961).

References

Bergstresser, K., Charnes, A., Yu, P. L. (1976). Generalization of domination structures and nondominated solutions in multicriteria decision making. *J. Optim. Theory Appl.* 18(1), pp. 3–13.

Chew, K. L. (1984). Maximal points with respect to cone dominance in Banach spaces and their existence. *J. Optim. Theory Appl.* 44(1), pp. 1–53.

Elster, K.-H., Gõpfert, A. (1987). Recent results on duality in vector optimization. In: Jahn, J., Krabs, W. (eds.): Recent advances and historical development of vector optimization. Springer-Verlag, Berlin-Heidelberg, pp. 129–136.

Gerth, C., Weidner, P. (to appear). Non-convex separation theorems and some applications in vector optimization. To appear in *J. Optim. Theory Appl.*

Hurwicz, L. (1958). Programming in linear spaces. In: Arrow, K. J., Hurwicz, L., Uzawa, H. (eds.): Studies in linear and nonlinear programming. Stanford University Press, Stanford, pp. 38–102.

Loridan, P. (1984). ε-solutions in vector minimization problems. *J. Optim. Theory Appl.*, 43(2), pp. 265–276.

Neumann, J. von, Morgenstern, O. (1961). Spieltheorie und wirtschaftliches Verhalten. Physica-Verlag, Würzburg.

Peleg, B. (1970). Efficiency prices for optimal consumption plans. *J. Math. Anal. Appl.* 29, pp. 83–90.

Radner, R. (1967). A note on maximal points of convex sets in l_∞. In: Lecam, L., Neyman, J. (eds.): Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1: Statistics. University of California Press, Berkeley and Los Angeles, pp. 351–354.

Weidner, P. (1983a). Vergleichende Darstellung von Optimalitätsbegriffen und Dualitätsansätzen in der Vektoroptimierung. Diplomarbeit, Martin-Luther-Universität Halle-Wittenberg.

Weidner, P. (1983b). Beziehungen zwischen Optimalitätsbegriffen in der Vektoroptimierung. Martin-Luther-Universität Halle-Wittenberg, Preprint der Sektion Mathematik, Nr.84.

Weidner, P. (1985a). Dominanzmengen und Optimalitätsbegriffe in der Vektoroptimierung. Wissenschaftl. Zeitschrift TH Ilmenau 31(2), pp. 133–146.

Weidner, P. (1985b). Charakterisierung von Mengen effizienter Elemente in linearen Räumen auf der Grundlage allgemeiner Bezugsmengen. Dissertation A, Martin-Luther-Universität Halle-Wittenberg.

Weidner, P. (1986a). Monotone Funktionale, Dualkegel und effiziente Elemente. Seminarbericht Nr. 80 der Sekt. Math., Humboldt-Universität zu Berlin, pp. 110–120.

Weidner, P. (1986b). Extensions of the feasible range leaving invariant the efficient point set. Seminarbericht Nr. 85 der Sekt. Math., Humboldt-Univ. Berlin, pp. 137–147.

Weidner, P. (1987a). The influence of neglecting feasible points and of varying preferences on the efficient point set. Wiss. Z. TH Ilmenau 33(6), pp. 181–188.

Weidner, P. (1987b). On interdependencies between objective functions and dominance sets in vector optimization. Seminarbericht Nr. 90 der Sekt. Math., Humboldt-Univ. Berlin, pp. 136–145.

Weidner, P. (1988a). Equivalent vector optimization problems and complete efficiency. Wiss. Schriftenreihe der TU Karl-Marx-Stadt 2, pp. 87–96.

Weidner, P. (1988b). On the characterization of efficient points by means of monotone functionals. *Optimization* 19(1), pp. 53–69.

Weidner, P. (1988c). Problems in models and methods of vector optimization. To appear in the proceedings of the conference "Multikriterielle Entscheidungen", Breitenbrunn, 1988, in Wiss. Schriftenreihe TU Karl-Marx-Stadt.

Weidner, P. (to appear). Complete efficiency and interdependencies between objective functions in vector optimization. To appear in: Zeitschrift für Operations Research.

Weidner, P. (in preparation). Preference relations and domination structures in vector optimization.

Wierzbicki, A. P. (1980). The use of reference objectives in multiobjective optimization. In: Fandel, G., Gal, T. (eds.): Multiple criteria decision making — theory and application. Lect. Notes Econ. Math. Syst. 177, pp. 468–486.

Yu, P. L. (1974). Cone convexity, cone extreme points and nondominated solutions in decision problems with multiobjective. *J. Optim. Theory Appl.* 14(3), pp. 319–377.

Dynamic Aspects of Multi-Objective Optimization

Andrzej P. Wierzbicki
Institute of Automatic Control
Warsaw University of Technology

Abstract

This paper is a review essay on the diverse possible understanding of dynamic aspects in multi-objective optimization. These aspects can be related either to various features of the optimization model or to the character of decision process. An optimization model with dynamic nature can lead either to multi-objective optimization of final state, or sequential multi-objective optimization, or multi-objective optimization of trajectories. If we include uncertainty issues, then their probabilistic, or fuzzy set, or set-valued system models are deeply related to multi-objective optimization and usually have also their dynamic aspects. A decision process is also multi-objective and dynamic in its essence, in all its main phases of intelligence (data acquisition), design (problem formulation) and choice (of actual decision). Especially important are the dynamic aspects of learning and changing preferences during the decision process. All these dynamic aspects of multi-objective optimization have diverse implications for current and future research.

1 Introduction

The subject of multi-objective dynamic optimization is of growing importance today. Wide spread computer applications, with new possibilities of advanced computer-men graphical and acoustic interaction, of computer networks and parallel computations, of symbolic model manipulation, create new dimensions and possibilities of multi-objective model analysis.

At the same time, there is a growing demand for new versions of decision support systems in business, management and industrial planning. Detailed and rigid central economic planning has been abandoned in most countries, since it cannot successfully deal with the complexity of modern civilization and economy. At the same time, most big enterprises entering world competition perform quite deep strategic analysis and long-term planning. Japanese firms attribute their successes to quite detailed but flexible long term planning — and point out that American firms tend to shorten too much their planning horizons.

Long-term, strategic planning requires comparisons of trajectories of changing decision attributes. This characterizes not only planning, but also predicting or generally analyzing future development. Thus, methods related to dynamic multi-objective optimization can be used in modern approaches to analyzing future development scenarios, as in the recent study of the Committee "Poland 2000" of P.Ac.Sc. (1989).

In order to cover in detail various dynamic aspects of multi-objective optimization and decision processes, a book rather than a short paper should be written. Even if we would restrict our attention to dynamic aspects of multi-objective optimization models, these aspects could be diverse. An optimization model can have dynamic nature described by difference equations, ordinary differential, partial differential, difference-differential, or integral equations. If the objectives correspond to only several functionals defined on model trajectories or functions of its final state, then we have the case of multi-objective optimization of final state, most widely covered in the literature.

Another case corresponds to a model using a difference equation with a multi-objective optimization problem defined at each stage — that is, to a sequential multi-objective optimization problem where the attention is focused on its multi-objective dynamic programming formalization. Such problems are rather intensively investigated in current literature.

Still another class of optimization models corresponds to the case when there is a rather large (or theoretically infinite) number of objectives but these objectives can be grouped into several objective trajectories (that, in turn, might correspond to solutions of difference, differential, difference-differential etc. equations). The results of infinite-dimensional multi-objective optimization are pertinent in such a case which, beside its theoretical interest, can have quite important practical implications.

Multi-objective optimization is also deeply related to the problem of decision uncertainty, which can be alternatively expressed either by probabilistic models, or fuzzy set models, or set-valued system models; all these formulations have their dynamic aspects.

On the other hand, multi-objective optimization can be also interpreted as a part of a broadly understood decision process which has multi-objective aspects in all its phases. Even if we restrict our attention to the classical phases of intelligence or information acquisition, of design or problem formulation, of choice or selection of an actual decision — see (Simon, 1958) — all these main phases of a decision process are not only multi-objective but also dynamic in nature, each with its own peculiarities.

Especially important are dynamic aspects of learning — both in the quantitative sense of acquiring more information pertinent for the decision and in the qualitative sense of becoming a master expert in a given field who intuitively and holistically processes all available information. Although some valuable results and models of learning have been proposed, this is a field that deserves much more attention in future research.

Since we want to comment — even if shortly — on all these aspects, the paper must have the character of an essay. On the other hand, we shall attempt at least to give an outline of mathematical models and results related to this important and broad field of research.

2 Final state of a dynamic model

It is well known in the theory of single-objective dynamic optimization that, under rather mild assumptions, optimizing a function of the final state of a dynamic model (called a Mayer problem) is equivalent to optimizing a functional defined along the trajectory of such a model (called a Lagrange problem) or to optimizing a sum of a function and a functional (called Bolza problem). This remains true — as we show further in more detail — if we consider multi-objective optimization of some functions of final state or some functionals defined along trajectories, provided their number remains finite.

For the sake of brevity, we shall consider here only two forms of dynamic models, described by ordinary differential or difference equations — while noting that similar conclusions can be reached for any other form. If the dynamic model is described by ordinary differential equations on a finite interval $[t_o;t_f]$, we can usually assume its nonsingular form of a standard system of *state equations*:

$$\dot{x}(t) = f(x(t), u(t), t), \quad t \in [t_o;t_f]; \quad x(t_o) = x_o \in R^n \tag{1a}$$

where *the state* $x(t) \in R^n$, while $f: R^n \times R^m \times R^1 \to R^n$ is a function that is continuously differentiable, at least in its first argument (additional assumptions e.g. on the bounds of Lipschitz constants might be needed to be certain that the solution of (1a) exists on all $[t_o;t_f]$). Its dependence on the *control* $u(t) \in R^m$ and on the *time* t might be only continuous, but differentiability is often assumed in order to apply gradient-like optimization techniques. Additionally, a *set of admissible control trajectories* $u = \{u(t), t \in [t_o, t_f]\}$ is defined:

$$\mathcal{U} = \left\{ u \in \overline{PC}([t_o;t_f], R^m) : \ u(t) \in U, \quad g(x(t), u(t), t) \leq 0 \in R^k \quad \text{a.a. } t \in [t_o;t_f] \right\} \tag{1b}$$

where $\overline{PC}([t_o;t_f], R^m)$ is the space of all limits of sequences of piece-wise continuous functions from $[t_o;t_f]$ into R^m. Broader spaces such as functions of bounded variation can be also assumed, but the essential point is to make the space complete which actually requires including not only functions but also distributions. $U \subset R^m$ is a bounded, usually compact set, expressing constraints on current values $u(t)$ of control, $g: R^n \times R^m \times R^1 \to R^k$ is a function of the same class as the function f, expressing additional, joint constraints on state and control (much more difficult to analyze than any constraints on $u(t)$ only), a.a. means for almost all t in the interval. Under such assumptions, the model (1a, b) defines a mapping $X: R^n \times \mathcal{U} \to R^n$ such that:

$$x(t_f) = X(x_0, u) \tag{2a}$$

whereas the *set of attainable final states* (starting from the initial state x_0) is defined as:

$$X_f(x_0) = X(x_0, \mathcal{U}) \tag{2b}$$

Let us additionally assume that a continuous (sometimes we might additionally require differentiability) function $h: R^n \to R^p$ is given whereas $y = h(x(t_f))$ is interpreted as optimized *outcome or objective vector*. Given x_0, the outcome vector y is thus determined as a function (actually, a vector functional) of control trajectories u:

$$y = F_0(u) = h(X(x_0, u)); \qquad F_0: \mathcal{U} \to R^p \tag{2c}$$

If some of the outcomes are separate functionals defined along both the control trajectory u and the entire state trajectory $x = \{x(t),\ t \in [t_0; t_f]\}$ — for example, in the form:

$$y_j = \int_{t_0}^{t_f} f_{0,j}(x(t), u(t), t)dt, \qquad j = 1, \dots p_f \tag{2d}$$

then it is sufficient — provided the functions $f_{0,j}$ are of the same class as functions f — to augment the state equations (1a) by p_f additional scalar equations of the form:

$$\dot{x}_{n+j}(t) = f_{0,j}(x(t), u(t), t), \quad t \in [t_o; t_f]; \quad x_{n+j}(t_0) = 0, \quad j = 1, \dots p_f \tag{2e}$$

and thus to convert the problem again to multi-objective optimization of a function of the final state; the same holds clearly if functionals (2c) have the form of Bolza, augmented by a continuous function of the final state.

In any case, all admissible $u \in \mathcal{U}$ define a finite dimensional *set of attainable outcomes* Y_0:

$$Y_0 = F_0(\mathcal{U}) \subset R^p \tag{2f}$$

In a sense, this is a static model (even if depending on entire control trajectories) defined with the help of a dynamic one; the specific form of the dynamic model influences only the details of optimization computations that we might perform with respect to outcomes y. In order to illustrate this point, consider an analogous model described by difference equations:

$$x(t+1) = f(x(t), u(t), t), \quad t = t_0, \dots t_f; \quad x(t_0) = x_0 \in R^n \tag{3a}$$

where the continuity of the function $f:\ R^n \times R^m \times R^1 \to R^n$ is sufficient for the existence of solutions of (3a) on any finite discrete time interval $[t_0, \dots t_f]$, provided the controls $u(t)$ are bounded; but we assume usually differentiability in order to apply gradient-like optimization techniques. The set of admissible controls is defined here similarly as in (1b):

$$\mathcal{U} = \left\{u \in B([t_0, \dots t_f], R^m):\ u(t) \in U,\ g(x(t), u(t), t) \le 0 \in R^k,\ t = t_0, \dots t_f\right\} \tag{3b}$$

where $B([t_0, \dots t_f], R^m)$ denotes the space of bounded functions from the discrete interval $[t_0, \dots t_f]$ into R^m. With these differences, the dynamic model (3a, b) defines again a mapping such as (2a) and, given an outcome function h, an outcome mapping (2b) and the set of attainable outcomes (2c).

Thus, the difference between the models (1a, b) and (2a, b) might influence the details of optimization techniques — which is often quite important: we shall recall that a maximum principle for optimizing a model (1a, b) does not require convexity assumptions because of the possibility of needle-like variations in time-continuous control that result in convexifying properties of integrals, while the results of maximum principle type for models (3a, b) are much weaker. However, this difference does not influence the essential formulation of a multi-objective optimization problem.

3 Basic concepts and parametric characterizations of multi-objective optimization

The sense of optimization in the outcome space R^p is defined by a positive cone C in this space. A standard form of the positive cone is:

$$C = \{y \in R^p : \; y_i \geq 0, \quad i = 1, \ldots p\} = R^p_+ \tag{4a}$$

and corresponds to the assumption that all outcomes or objectives are maximized; minimized objectives can be taken into account by changing their signs. Another example of positive cone:

$$C' = \{y \in R^p : \; y_i \geq 0, \quad i = 1, \ldots p_p; \; y_i = 0, \; i = p_p + 1, \ldots p\} \tag{4b}$$

represents less standard assumption that first p_p outcomes are maximized while remaining ones are stabilized around a given reference level, i.e. maximized when below this level but minimized when above it. Note that both C and C' are pointed cones, the only subspace they contain is the trivial one $\{0\}$ — which we shall require from any positive cone. They are also closed convex cones, which is an important but not essential property of a positive cone; finally, C has nonempty interior while $int\ C' = \emptyset$ (an empty set).

Given a positive cone C, we define the set of efficient outcomes (called Pareto-optimal if $C = R^p_+$) in a standard way:

$$\hat{Y}_0 = \{\hat{y} \in Y_0 : \; Y_0 \cap (\hat{y} + \tilde{C}) = \emptyset\}; \qquad \tilde{C} = C \setminus \{0\} \tag{5a}$$

where $\hat{y} + \tilde{C}$ denotes the cone $\tilde{C}$ — that corresponds to a strict inequality in the outcome space — shifted by the vector $\hat{y}$. A strong inequality in the vector space corresponds to the cone $int\ C$ (at least, if $C = R^p_+$); thus, weakly efficient (or weakly Pareto-optimal) outcomes are defined by:

$$\hat{Y}_0^w = \{\hat{y} \in Y_0 : \; Y_0 \cap (\hat{y} + int\ C) = \emptyset\} \tag{5b}$$

The concept of weakly efficient outcomes is typically too weak for applications (note that $\hat{Y}_0^w = Y_0$ if we use the cone C'), but they are used in the theory because it is simpler to prove some theorems for them. Most useful for applications is the concept of properly efficient outcomes that can be defined in various ways — see (Kuhn and Tucker 1950, Geoffrion, 1968, Henig, 1982, Sawaragi et al., 1985, Wierzbicki, 1977, 1986, 1990). We shall use here one of the latter definitions of properly efficient outcomes with a prior bound ε (such that the corresponding trade-off coefficients have a bound that is a priori known, approximately $1/\varepsilon$, as opposed to a bound that exists but we do not know how large it is). For this purpose, we define first the closure of an ε-conical neighborhood of the cone C:

$$C(\varepsilon) = \{y \in R^p : \; dist\ (y, C) \leq \|y\|\} \tag{6a}$$

where any norm in R^p can be used and the distance of y from C is defined as a Haussdorf distance that uses a topologically equivalent norm (hence, in R^p, it can be any other norm). If $C = R^p_+$, a closed convex cone $C(\varepsilon)$ with useful properties is obtained if the norm l_1 is used on the right-hand side and the norm l_∞ augmented with l_1 multiplied by 2ε is used to define the distance, see (Wierzbicki, 1990). In any case, properly efficient outcomes with prior bound ε can be defined as:

$$\hat{Y}_0^{p\varepsilon} = \{\hat{y} \in Y_0 : \; Y_0 \cap (\hat{y} + int \; C(\varepsilon)) = \emptyset\} \tag{6b}$$

Note that the cone $C(\varepsilon)$ has a nonempty interior even if C has an empty interior, as in the case of C'; thus, it does not actually matter that properly efficient outcomes with a prior bound are defined similarly to weakly efficient ones, because we use a broader cone in their definition. The set of properly efficient outcomes in their traditional sense, without a prior bound (that is, with only an existential bound) is then defined as:

$$\hat{Y}_0^p = \bigcup_{\varepsilon > 0} \hat{Y}_0^{p\varepsilon} \tag{6c}$$

Note that we have defined the three types of efficiency (5a), (5b) and (6b) rather abstractly, except for the specific examples of positive cones; thus, these definitions can be used as well in infinite dimensional outcome spaces. However, if the space of outcomes is finite dimensional and directly related to the space of final states of a dynamic model, there is not much difference between dynamic and static multi-objective optimization: we can use all known multi objective optimization theory for a finite dimensional set Y_0, and first thereafter take into account the dynamic character of the underlying model.

There are two basic types of questions in multi-objective optimization. One is to analyze the entire sets $\hat{Y}_0$, $\hat{Y}_0^w$ or $\hat{Y}_0^{p\varepsilon}$; but we can usually analyze only their general, qualitative properties. Another is to select some representative $\hat{y} \in \hat{Y}_0$ ($\hat{Y}_0^w$ or $\hat{Y}_0^{p\varepsilon}$) for a human decision maker to choose from. Historically, a third type of questions has been also analyzed: to find a universal way of selection of $\hat{y} \in \hat{Y}_0$ ($\hat{Y}_0^w$ or $\hat{Y}_0^{p\varepsilon}$) that would make the choice by a human decision maker unnecessary. However, it was rather soon realized that such a decision automation is good for machines only, in repetitive, standard decision problems — and even then there is no universal way of selecting $\hat{y} \in \hat{Y}_0$ independently of particular decision situation.

In both (or all three) of these types of problems, a general method of analyzing efficient outcomes is to introduce a *parametric scalarizing function*. A set of *controlling parameters* $W \subset R^p$ is defined; examples of such parameters are *weighting coefficients*, actually the elements of the dual space to the space of outcomes (which distinction is not very essential in a finite dimensional case) or *reference points*, that can be interpreted as *aspiration* or *reservation levels* for objectives and are the elements of the primal space. A parametric scalarizing function is a continuous function $s : R^p \times W \to R^1$ that possesses the following *sufficiency property*: there is a nonempty set $W^s \subset W$ such that:

$$\hat{\Psi}(w) = \text{ Arg } \max_{y \in Y_0} s(y, w) \subset \hat{Y}_0 \;\text{ for all } w \in W^s \tag{7a}$$

where Arg max denotes the set of maximal points of the function. We might substitute $\hat{Y}_0$ in (7a) by $\hat{Y}_0^w$ or $\hat{Y}_0^{p\varepsilon}$ (and W^s by W^{sw} or $W^{sp\varepsilon}$), thus obtaining a slightly weaker or stronger sufficiency property. Theoretically, the set W^s might depend on Y_0; however, we shall see that for a broad class of scalarizing functions the corresponding sets W^s are actually independent of Y_0. There are also some more complicated ways of introducing parametric scalarization in multi-objective optimization for which additional constraints of the form $y \in Y_{as}(w) \cap Y_0$ are assumed in (7a) or even a requirement of repeating the maximization p times is added.

Without such modifications, a scalarizing function has the sufficiency property if it is monotone in an appropriate sense. If W^s denotes the set of such w that $y'' - y' \in \tilde{C} = C \setminus \{0\}$ implies $s(y'', w) > s(y', w)$ (such w that the function $s(., w)$ is strongly monotone), then (7a) holds. If W^{sw} denotes the set of such w that $y'' - y' \in int\ C$ implies $s(y'', w) > s(y', w)$ (such w that the function $s(., w)$ is strictly monotone), then (7a) holds for $\hat{Y}_0^w$, W^{sw}. If W^{spe} denotes the set of such w that $y'' - y' \in int\ C(\varepsilon)$ implies $s(y'', w) > s(y', w)$ (such w that the function $s(., w)$ is $C(\varepsilon)$-strictly monotone — which implies strong monotonicity), then (7a) holds for $\hat{Y}_0^{pe}$, W^{spe} — see e.g. (Wierzbicki 1986, 1990).

A parametric scalarizing function has also *necessity property* that might be either *complete* or *incomplete*. An incomplete necessity property is obtained if for a given set $W^n \subset W$ there exists a nonempty subset $\tilde{Y}_0 \subset \hat{Y}_0$ such that, if $\hat{y} \in \tilde{Y}_0$, then there exists $\hat{w} \in W^n$ for which:

$$\hat{y} \in \hat{\Psi}(\hat{w}) = \text{Arg} \max_{y \in Y_0} s(y, \hat{w}) \tag{7b}$$

Again, we can substitute $\hat{Y}_0$ by $\hat{Y}_0^w$ or $\hat{Y}_0^{pe}$ (and $\tilde{Y}_0$ by $\tilde{Y}_0^w$ or $\tilde{Y}_0^{pe}$, W^n by W^{nw} or W^{npe}) to obtain incomplete necessity property for weakly efficient or properly efficient outcomes with prior bound. Since a scalarizing function is supposed to have the sufficiency property, we can always take $W^n = W^s$ and obtain a nonempty $\tilde{Y}_0$ in the above definition (with appropriate modifications for the cases of weakly or properly efficient outcomes). However, we might also choose different, usually larger sets W_K^n (W_K^{nw} or W_K^{npe}) in order to obtain larger sets $\tilde{Y}_0$ ($\tilde{Y}_0^w$ or $\tilde{Y}_0^{pe}$) that desirably should cover $\hat{Y}_0$ ($\hat{Y}_0^w$ or $\hat{Y}_0^{pe}$), in which case it is said that the necessity property is complete. If it is complete and $W^{nw} = W^{sw}$, or $W^{np} = W^{sp}$, or — equivalently — if:

$$\bigcup_{w \in W^{spe}} \hat{\Psi}(w) = \hat{Y}_0^{pe} \tag{7c}$$

(similarly for W^{sw} and $\hat{Y}_0^w$), then we say that the scalarizing function $s(y, w)$, used when defining $\hat{\Psi}(w)$, *completely characterizes parametrically the set of properly efficient outcomes with prior bound* ε (or, with obvious modifications, the set of weakly efficient outcomes).

The use of the term "characterizes" is justified because, if (7c) holds, we can use the maximization of the scalarizing function not only as a sufficient, but also a necessary condition of proper efficiency with prior bound ε: for every outcome in $\hat{Y}^{pe}$ there exists a parameter vector $\hat{w} \in W^{spe}$ such that this outcome maximizes the scalarizing function $s(y, \hat{w})$. We did not define a complete characterization for (strictly) Pareto-optimal outcomes, because it is known in the theory of multi-objective optimization — see e.g. (Sawaragi et al., 1985) or an impossibility theorem in (Wierzbicki, 1986) — that the set of efficient or Pareto-optimal outcomes, without its prior knowledge nor repeated maximizations, can be only *almost completely characterized* — in such a way that:

$$\hat{Y}_0 \subset closure \bigcup_{w \in W^s} \hat{\Psi}(w) \tag{7d}$$

Beside completeness, a characterization of efficient solutions by maxima of a scalarizing function can have various other properties — see (Wierzbicki, 1986). Such a characterization is *parametrically* (locally) *controllable* if the (point-to-set) mapping $\hat{\Psi}(w)$ is Lipschitz-continuous (with a Haussdorf distance defining $\|\hat{\Psi}(w'') - \hat{\Psi}(w')\|$). It is *independent of prior information* if W^s (W^{sw}, W^{spe}) does not depend on a prior knowledge about the shape of $\hat{Y}_0$ ($\hat{Y}_0^w$, $\hat{Y}_0^{pe}$). We might use also more descriptive properties of such parametric characterizations — their easy computability, interpretability of their parameters w or parametric selections $\hat{\Psi}(w)$, etc.

4 Classes of scalarizing functions and their use in multi-objective optimization of final state

There are several important classes of scalarizing functions. If $C = R_+^p$, the most elementary is the (bi-)linear function or the weighted sum of objectives:

$$s(y, w) = \Lambda^T y = \sum_{i=1}^{p} \Lambda_i y_i; \qquad w = \Lambda \tag{8a}$$

with:

$$W^s = int\ R_+^p = W^{sp} = W^{np}; \qquad W^n = R_+^p \setminus \{0\} = W^{sw} = W^{nw} \tag{8b}$$

where we often additionally normalize the values of weighting coefficients (under an implicit assumption that the objective outcome values are also normalized, see later discussion):

$$\bar{\Lambda}_i = \Lambda_i / \sum_{j=1}^{p} \Lambda_j \tag{8c}$$

If the attainable outcome set Y_0 is convex, then the weighted sum completely characterizes the weakly Pareto-optimal outcomes (with $W^{sw} = W^{nw} = W^n$ which means that weighting coefficients are nonnegative and not all equal zero), the properly Pareto-optimal outcomes (with $W^{sp} = W^{np} = W^s$ which means that weighting coefficients are positive) and the properly Pareto-optimal outcomes with a prior bound ε (with additional restriction that $\bar{\Lambda}_i \geq \varepsilon/(1 + p\varepsilon)$ which results in trade-off coefficients bounded by $1 + 1/\varepsilon$, see (Wierzbicki, 1990). The (strictly) Pareto-optimal outcomes are only almost completely characterized with $W^n \neq W^s$ but $W^n \subset$ *closure* W^s. However, the main deficiencies of weighted sum scalarization are that its necessary conditions hold only under convexity assumptions (since they rely on arguments on separating convex sets by linear functions; the sufficient conditions rely on monotonicity and are thus independent of convexity) and that this scalarization is not parametrically controllable even in the simplest case when the set Y_0 is a convex polyhedron.

The scalarization by a weighted sum is deeply related to control theory with multiple objectives — or to the multi-objective optimization of final state. A basic property of dynamic models (1a) with continuous time is that their sets of attainable final states $X_f(x_0)$ are convex for all x_0 whenever the set of admissible control trajectories (1b) is *operationally convex*. The concept of operational convexity is much weaker than the concept of simple convexity and means that if two controls $u', u'' \in \mathcal{U}$, they are admissible, then also admissible is their arbitrary needle-like variational convex combination. This combination is obtained as follows: we subdivide the interval $[t_0; t_f]$ into arbitrarily short subintervals of length $\delta\Delta t$, $(1-\delta)\Delta t$ and let the combined control $u(t)$ be equal to $u'(t)$ on subintervals $\delta\Delta t$ and to $u''(t)$ on subintervals $(1-\delta)\Delta t$; then we take the limit as $\Delta t \to 0$, whereas the combined control ceases to be a function but becomes a Cesari-type distribution, a limit of a sequence of piece-wise continuous functions. Note that the set (1b) is operationally convex whenever the joint constraints $g(x(t), u(t), t) \leq 0$ for state and controls are either absent or linear, no matter whether U is convex or not.

The basic property of dynamic models with continuous time variable — that their set of attainable final states, $X_f(x_0)$ is convex even for nonlinear models and independently on x_0 whenever $\mathcal{U}$ is operationally convex — is also called the convexifying property of integrals. This property is also the basis of various proofs of maximum principle. But Chang (1966) used it to prove one of the first versions of theorems on multi-objective optimization of final state, so called general optimal control theorem. Using the concepts of the present paper, this theorem means that a weighted sum (almost) completely characterizes Pareto-optimal outcomes related to the final state of a time-continuous dynamic model, if the function h is linear and the set $\mathcal{U}$ is operationally convex; moreover, a corresponding form of maximum principle can be used to determine the multi-objectively optimal control.

Another basic class of scalarizing functions are norms of distance from a point in Y_0 and a shifted utopia or ideal point, defined as any point $\bar{y} \in \bar{Y} \subset R^p$ that dominates entire Y_0:

$$\bar{Y}_0 = \{\bar{y} \in R^p : \; Y_0 \subset \bar{y} - C\} \tag{9a}$$

If $C = R^p_+$, then $\bar{Y}_0 = \bar{\bar{y}} + C$, where $\bar{\bar{y}}$ is so called *utopia* or *ideal point* obtained by maximizing subsequently all objectives and combining their maxima into one vector. When looking for a universal way to select $\hat{y} \in \hat{Y}_0$, the following "parameter-free" scalarizing function was suggested:

$$s(y) = \|\bar{y} - y\|, \qquad \bar{y} = \bar{\bar{y}} \text{ or } \bar{y} \in \bar{Y}_0 \tag{9b}$$

It is easy to see that $\bar{y} \in \bar{Y}_0$ with $C = R^p_+$ implies at least strict monotonicity of this function — hence all its maxima are at least weakly Pareto-optimal — while for many norms, such as the norms l_p with $1 \leq p < \infty$, $\bar{y} \in int\ \bar{Y}_0$ implies strong monotonicity, hence all maxima are Pareto-optimal in this case. However, the function (9b) in fact is not parameter-free, even if we use $\bar{y} = \bar{\bar{y}}$.

In order to use a norm in the outcome space it is necessary that various outcomes are comparable and can be summed. This would be a very strong restriction since typically various outcomes have quite different meaning and physical units. Thus, when writing a norm such as (9b) we implicitly assume that the *outcomes are normalized*: for each outcome y_i there is an interval of bounds $[y_i^{\text{low}}; y_i^{\text{upp}}]$ given and the outcome transformed to a dimensionless normalized outcome y_i^n:

$$y_i^n = y_i/(y_i^{\text{upp}} - y_i^{\text{low}}) \tag{9c}$$

But there is no standard, "objective" way to define intervals of bounds. Even if we accept $\bar{\bar{y}}$ as a "natural" upper bound, which is also open to discussion, a "natural" definition of a lower bound is much more difficult. Some researchers assume that all objective outcomes are positive and thus the lower bound is zero; but such assumption is also arbitrary, since adding a positive constant to a bounded function can always make it positive, the question is how much to add?

Other researchers used so called *nadir point* that is supposed to represent a tight lower bound on efficient outcomes — the maximal element between such $\bar{y}$ that $\hat{Y}_0 \subset \bar{y} + C$. However, computing a tight lower bound on $\hat{Y}_0$ is a rather difficult problem and simplistic approaches to it lead to inaccurate results. Any approximation of the nadir point, but even the selection of the nadir as a lower bound, is in fact subjective. On the other hand, if we have such an approximation, use it as a lower bound together with utopia point as an upper bound to normalize all outcomes, and minimize the function $s(y) = \|\bar{y}^n - y^n\|$, we obtain a Pareto-optimal outcome that is in some sense *neutral* and thus is useful to start analysis of other Pareto-optimal outcomes. But the sense of neutrality of this outcome is subjective, depends on the selection of upper and lower bounds and on the norm used. Thus, we cannot attach a general significance to such an outcome — although we can, for example, examine its relations to various concepts of cooperative solutions in game theory.

In general, *we should not try to prescribe* a selection between efficient outcomes, but leave it to actual decision maker — and a parameterization of a scalarizing function should be helpful in proposing various efficient outcomes to him. The arbitrariness of normalization (9c) is in fact equivalent to choosing some weighting coefficients for all outcomes — although it makes obviously more sense to use normalized weighting coefficients (8c) first after normalizing all outcomes to a common range. Thus, we can define:

$$s(y,w) = \|\bar{y}^n - y^n\|; \qquad y_i^n = \bar{\lambda}_i y_i/(y_i^{\text{upp}} - y_i^{\text{low}}) \tag{9d}$$

with $w = \Lambda \in W^s = int\ R_+^p$; but the details of parametric characterization with help of this function depend on the norm used. If we take any norm l_p with $1 \le p < \infty$, we obtain an almost complete characterization of (strictly) Pareto-optimal outcomes, provided $\bar{y} \in \bar{Y}_0$. If we assume $\bar{y} \in int\ \bar{Y}_0$ and take the Chebyshev norm l_∞, we obtain a complete characterization of weakly Pareto-optimal outcomes; with the same $\bar{y}$, if we take an augmented Chebyshev norm (the sum of the norm l_∞ and the norm l_1 multiplied by ε), we obtain a complete characterization of properly Pareto-optimal outcomes with prior bound ε. These characterizations do not depend on convexity assumptions.

The scalarizing functions constructed with norms of the distance from an ideal or utopia outcome were first introduced by researchers in the U.S.S.R., see e.g. (Volkovich, 1969), for static multi-objective optimization. Salukvadze (1971), (1979) used them for dynamic multi-objective optimization of final state, while hoping to obtain this way an "objective", "parameter-free" solution (he assumed positive values of all objective outcomes without analyzing more deeply this assumption). The results of Salukvadze, including several versions of maximum principle, must be considered as one of the first in dynamic multi-objective optimization. They were popularized and extended in the U.S.A. first by Yu (1972), then by Zeleny (1973, 1982) and others — but adapted again for the static case, while the originators of this idea remained almost unknown in the West.

Without requiring that $\bar{y} \in \bar{Y}_0$ or $\bar{y} \in int\ \bar{Y}_0$, a scalarizing function of the form (9d) has been used also extensively in so called *goal programming* approaches — see among others

(Charnes and Cooper, 1975), (Ignizio, 1978). Goal programming approaches actually use both weighting coefficients and the vectors $\bar{y}$ (interpreted as *goals*) as parameters in the scalarizing function. They have been also used for multi-objective trajectory optimization of time-discrete dynamic models, see further discussion. However, if we do not assume $\bar{y} \in \tilde{Y}_0$, it is rather difficult to obtain efficient solutions when minimizing (9d) — we need then repeated optimizations, additional constraints and convexity assumptions to check efficiency, or we must switch from minimizing to maximizing (9d) with an attainable $\bar{y} \in Y_0$ and additional constraints, see e.g. (Wierzbicki, 1986).

A scalarizing method that relies exclusively on parameters $\bar{y}$ — interpreted as *reservation levels* — is the method of constraint perturbations. We define the j-th *perturbation function* of Pareto-optimal outcomes as:

$$\tilde{h}_j(\bar{y}) = \max_{y \in \tilde{Y}_0^{(j)}(\bar{y})} y_j \tag{10a}$$

where:

$$\tilde{Y}_0^{(j)}(\bar{y}) = \{y_K \in Y_0 : \; y_i \geq \bar{y}_i \text{ for all } i \neq j\} \tag{10b}$$

It is known — see (Geoffrion, 1971, Haimes et al., 1974, Benson et al., 1977, Sawaragi et al., 1985) — that an outcome $\hat{y} \in Y_0$ is (strictly) Pareto-optimal, $\hat{y} \in \hat{Y}_0$, if and only if with $\bar{y} = \hat{y}$ we obtain $\tilde{h}_j(\hat{y}) = \hat{y}_j$ when solving problems (10a) for all $j = 1, \ldots p$. If Y_0 is convex, an outcome $\hat{y} \in Y_0$ is properly Pareto-optimal (without a prior bound), $\hat{y} \in \hat{Y}_0^p$, if and only if additionally the problems (10a) are *stable* for all $j = 1, \ldots p$ — that is, the functions $\tilde{h}_j(\bar{y})$ are Lipschitz-continuous with respect to $\bar{y}$ or, equivalently, the subdifferentials of these functions or the sets of Lagrange multipliers for the inequalities in (10b) are bounded.

This characterization, although it is complete for (strictly) Pareto-optimal outcomes without convexity assumptions, it is by no means satisfactory. If we do not know that $\bar{y} \in Y_0$, the sets (10b) might become empty, which causes problems in computations. This characterization requires p repetitions of maximization; without such repetition, we are sure to obtain this way only weakly Pareto-optimal outcomes. Even if we repeat the maximization p times, the problems (10a) are unstable for such Pareto-optimal outcomes that are not properly Pareto-optimal — which means that the corresponding computational problems are badly conditioned, thus it is not really practical to check this way Pareto-optimality of such outcomes. Actually, the only practical way of checking Pareto-optimality of such outcomes is theoretical — by closure arguments. The above characterization of properly Pareto-optimal outcomes, though very important theoretically, is restricted to convex problems; moreover, it does not provide a constructive way of checking computationally whether the problems (10a) are stable.

Thus, the method of constraint perturbation is not easily adaptable for multi-objective optimization of final state, when the requirements of repeating rather complex optimization calculations many times and the necessity of using attainable $\bar{y}$ are rather restrictive. This motivated Wierzbicki (1975) to remove such requirements by using penalty methods when solving problems (10a) for dynamic multi-objective optimization cases. A modification of the problems (10a) with a differentiable penalty function is:

$$\tilde{h}_j^{\rho}(\bar{y}) = \max_{y \in Y_0} s_j(y, \bar{y}, \rho) \tag{11a}$$

where:

$$s_j(y,\bar{y},\rho) = y_j + 0.5\rho\sum_{i\neq j}(\min(0,y_i-\bar{y}_i))^2 \tag{11b}$$

with some penalty coefficient $\rho > 0$. We have thus replaced constraints with penalty terms, interpreted as soft constraints that might be violated; therefore, $\bar{y}$ need not be attainable. Moreover, precisely when these soft constraints are violated at a maximal point $\hat{y}$ of $s_j(.,\bar{y},\rho)$, $\hat{y}_i < \bar{y}_i$ for all $i \neq j$, then $\hat{y} \in \hat{Y}_0$, it is (strictly) Pareto-optimal — as a result of a single, not repeated maximization — because the function $s_j(.,\bar{y},\rho)$ is then strongly monotone. Thus, we can use such penalty functions effectively when computing Pareto-optimal solutions for dynamic models.

This does not help much, however, when checking the Pareto-optimality of outcomes that are not properly Pareto-optimal. The penalty function (11a, b) has the property that its maximal points correspond to stable problems (10a) with bounded Lagrange multipliers for the inequalities in (10b) — see e.g. (Wierzbicki, 1984). Thus, all its maxima are properly Pareto-optimal if $\hat{y}_i < \bar{y}_i$ for all $i \neq j$; the penalty function would simply stubbornly refuse to find a maximum at an improperly Pareto-optimal outcome, would find instead a properly Pareto-optimal outcome that is close to the improperly Pareto-optimal one.

If we tried to use an exact nondifferentiable penalty function when substituting the squared terms in (11b) by absolute values, it would also refuse to find maxima at improperly Pareto-optimal outcomes. This suggests, however, a practical way of checking the stability of the problems (10a, b) and the proper Pareto-optimality — at least, with a prior bound — of related outcomes without convexity assumptions. This is also related to another issue: how to construct, with help of a norm, a scalarizing function that would characterize parametrically properly efficient outcomes while using a *reference point* $\bar{y}$ as a parameter not restricted to $\bar{y} \in \hat{Y}_0$ nor to $\bar{y} \in Y_0$ — in other words, how to remove the disadvantages of both shifted ideal and goal programming methods.

The construction of such a function is related to the concepts of the cone $C(\varepsilon)$ and of separating the interior of this cone from Y_0 by a nonlinear function. If $C = R^p_+$ and we use the l_1 norm on the right-hand side of the inequality in (6a) and the l_∞ norm augmented with l_1 to define *dist* (y, C), then we obtain a cone $C(\varepsilon) = C(\varepsilon, l_1, l_\infty)$:

$$C(\varepsilon,l_1,l_\infty) = \{y \in R^p : \ \|y^{(-)}\|_{l_\infty} + 2\varepsilon\|y^{(-)}\|_{l_1} \leq \|y\|_{l_1}\} \tag{12a}$$

where:

$$y^{(-)} = (\min(0,y_1),\ldots\min(0,y_i),\ldots\min(0,y_p)) \tag{12b}$$

This cone can be equivalently written in several forms, such as:

$$\begin{aligned} C(\varepsilon,l_1,l_\infty) &= \{y \in R^p : \ -y_j \leq \varepsilon\sum_{i=1}^{p} y_i, \ \ j = 1,\ldots p\} = \\ &= \{y \in R^p : \ \min_{1\leq i\leq p} y_i + \varepsilon\sum_{i=1}^{p} y_i \geq 0\} \end{aligned} \tag{12c}$$

The first of these forms implies that $C(\varepsilon, l_1, l_\infty)$ is a convex polyhedral cone. The second indicates that the interior of this cone can be strictly separated from the rest of the space by a nonlinear and nondifferentiable, but continuous and monotone function; by appropriately shifting this function, we can also separate a shifted cone $\bar{y} + int\ C(\varepsilon)$.

The concept of separation is very strong: the statement that a function strictly separates, at some point $\bar{y} \in Y_0$, the set Y_0 and the shifted cone $\bar{y} + int\ C(\varepsilon)$, is equivalent to three statements: (i) that $\bar{y}$ maximizes this function over $y \in Y_0$, (ii) that $\bar{y} \in \hat{Y}_0^{p\varepsilon}$, it is properly efficient with prior bound ε, and (iii) that the function is $C(\varepsilon)$-strictly monotone (at least at the point $\bar{y}$). With help of this equivalence, it can be shown — see (Wierzbicki, 1990) — that the following scalarizing function:

$$s(y, w) = \min_{1 \leq i \leq p} (y_i - \bar{y}_i) + \varepsilon \sum_{i=1}^{p} (y_i - \bar{y}_i); \quad w = \bar{y} \in W^s = R^p \tag{13a}$$

completely characterizes, without any convexity assumptions, properly Pareto-optimal outcomes with a prior bound ε — and that trade-off coefficients are in this case indeed bounded by $1 + 1/\varepsilon$. Moreover, a single maximization of this function is equivalent to solving p times modified perturbation problems (10a, b):

$$\tilde{h}_j(\bar{y}) = \max_{y \in \tilde{Y}_0^{(j)\varepsilon}(\bar{y})} \Big(y_j + \varepsilon \sum_{i=1}^{p} y_i \Big) \tag{13b}$$

where:

$$\tilde{Y}^{(j)\varepsilon}(\bar{y}) = \Big\{ y \in Y_0 : \ y_i \geq \bar{y}_i + \varepsilon \sum_{k=1}^{p} (\bar{y}_k - y_k) \ \text{ for all } i \neq j \Big\} \tag{13c}$$

Therefore, by maximizing the function (13a), we check also the stability of perturbation problems (11a, b) — which are limits of (13b, c) as $\varepsilon \to 0$ — and this checking is independent from convexity assumptions. Naturally, the function (13a) and the problems (13b, c) are formulated assuming that the objective outcomes are earlier normalized — we omit here the notation y^n, $\bar{y}^n$ just for simplicity of corresponding expressions.

The scalarizing function (13a) is an example of order-consistent achievement scalarizing functions, see (Wierzbicki, 1986). This class of scalarizing functions is most useful for all cases of multi-objective optimization of time-discrete dynamic models when the convexity of the set of attainable outcomes cannot be taken for granted.

For example, for discrete-time dynamic models of the type (3a, b), we can be sure of the convexity of the set of attainable outcomes defined similarly as in (2a, b, c, f) only if the functions f, g, h are linear and the set U is convex. Even in this case, it is better to use the achievement scalarizing function (13a) — which maximization can be then rewritten as a dynamic linear programming problem — than the weighted sum (8a), since the achievement scalarizing function (13a) results in parametrically controllable characterization. If the functions f, g, h are nonlinear and the set of attainable outcomes might be nonconvex, the scalarizing function (13a) has the advantage that it still completely characterizes properly Pareto-optimal solutions with prior bound. However, in order to effectively solve the problem of its maximization in such a case, we have either to employ time-discrete nondifferentiable optimization techniques — see (Clarke, 1983, Kiwiel, 1985) — or to use differentiable approximations, see (Kreglewski et al., 1989).

5 Multi-objective dynamic programming

Let us further consider the discrete-time dynamic model (3a, b) while explicitly stating assumptions about decision outcomes; for an optimization of the final state, the outcomes y are defined by the equation:

$$y = h(x(t_f)) \tag{14a}$$

where $h: R^n \rightarrow R^p$ is a continuous (or differentiable) function. Note that we could assume that y depends also on all previous $x(t)$, $u(t)$ for $t = t_0, \ldots t_f - 1$, but such a dependence can be rewritten to the form (14a) through a suitably extended definition of $x(t_f)$. Note also that, given a control trajectory $u = \{u(t_o), \ldots u(t_f - 1)\}$, any solution $x(\tau)$ of (3a) can be expressed as a transition function:

$$x(\tau) = X(\tau, t, x(t), u[t, \ldots \tau - 1]) \tag{14b}$$

where $\tau \geq t + 1$ and $u[t, \ldots \tau - 1] = \{u(t), \ldots u(\tau - 1)\}$ is the restriction of the control trajectory u to the discrete subinterval $[t, \ldots \tau - 1]$, while we can interpret (2a) as a shorthand notation for (14b) with $t = t_0$, $\tau = t_f$.

Suppose the sense of optimization in the outcome space is defined by a positive cone $C = R^p_+$ as in (4a), (5a). Together with (14b), this leads to the question whether the optimality principle and the resulting dynamic programming technique known for single-objective dynamic optimization can be extended to the multi-objective case.

The answer is positive under additional assumptions that can be stated in various ways, but are generally related to separability and monotonicity. It is clear that in order to use the basic argument of the principle of optimality we must be sure — see e.g. (Wierzbicki, 1984) — of state-transition properties of the dynamic model, which in the case of the model (3a, b) is guaranteed by its property (14b), and of separability of control in time: given two admissible control trajectories u' and u'', each their separable combination u such that $u(\tau) = u'(\tau)$, $\tau < t$, $u(\tau) = u''(\tau)$, $\tau \geq t$ for any given $t = t_0, \ldots t_f - 1$ must be also admissible. Clearly, with the set of admissible controls defined as in (3b), the separability of control in time requires either that the constraining function g is independent of $x(t)$ or that it has rather specific form.

If we have the property (14b) and the separability of control in time, then an *optimality principle for multi-objective optimization of final state* is immediate: given any Pareto-optimal outcome $\hat{y}$ and the corresponding control and state trajectories $\hat{u}$ and $\hat{x}$, for any given $t = t_0, \ldots t_f - 1$, the restriction $\hat{u}[t, \ldots t_f - 1]$ must be Pareto-optimal for a problem starting with $\hat{x}(t)$, t, because otherwise $\hat{u}$ could not be Pareto-optimal as a whole control trajectory. If the outcomes y in (14a) depend also on previous $x(t)$, $u(t)$ — which might be a reasonable assumption when the elimination of such dependence would require a too large increase of the dimensionality of state — then additional assumptions concerning the monotonicity of such dependence are needed, see e.g. (Li and Haimes, 1987), although the question of a most natural form of such assumptions is still open. This issue and other questions related to so-called multi-objective dynamic programming received recently much attention and motivated many papers, particularly in the U.S.A., see (Li and Haimes, 1988) for a review.

Here we must warn the reader that in such papers he will often encounter the mental shortcut of implicitly but incorrectly presenting the concept of ***dynamic programming*** as equivalent to ***dynamic optimization***. This is a rather misleading interpretation, since:

(a) historically, dynamic optimization is a much older and broader subject than dynamic programming (especially in Eastern Europe and in the U.S.S.R., but this is true also in the U.S.A., where the calculus of variations developed many methods of dynamic optimization by the works of M. Hestenes and many other researchers);

(b) the particular technique called dynamic programming as proposed by S. Dreyfus and further extended and popularized by R. Bellman (together with various forms of optimality principle, Bellman-Hamilton-Jacobi type of variational equations, diverse variants of using optimal value function when computing optimal control) is very powerful theoretically but not necessarily effective as a computational tool;

(c) specifically, this particular technique is good as a computational tool for problems with a small, discrete number of admissible states of a dynamic process, but a very poor computational tool whenever the number of admissible states is large — in particular, if admissible states belong to a set of continuum power, such as in the model (3a, b);

(d) when speaking about the "curse of dimensionality" — that is, the exponential growth of computational effort with the dimensionality of the state space — R. Bellman did not mention that it is a feature of only rather simplistic approaches to dynamic optimization that he used as a comparison to the dynamic programming technique (which has also exponential, although slower increase of computational effort with the dimensionality of state space);

(e) even before the dynamic programming technique, other techniques of dynamic optimization were known that — although they require stronger assumptions and thus are less general than the dynamic programming technique — can be shown today to have only polynomial dependence on the dimensionality of the state space. This has been only recently recognized in the U.S.A. due to the celebrated dispute over the "algorithm" of L. Khachian (but the dispute was similarly misinformed — the algorithm was devised and widely used by N. Shor and many other researchers, particularly from Kiev group, while L. Khachian used it only to prove the polynomial dependence of computational effort on the dimensionality).

With all these reservations, multi-objective dynamic programming is certainly a subject worth further research — at least, as a powerful theoretical tool of increasing understanding — but should not be considered as a key to all problems of multi-objective dynamic optimization.

6 Multi-objective trajectory optimization

Here we consider again the discrete-time dynamic model (3a,b) but substantively change the assumptions about decision outcomes:

$$y(t) = h(x(t), t) \tag{15a}$$

where $h : R^n \times R^1 \to R^p$ is a continuous (or differentiable) function. Suppose for each $t = t_0, \dots t_f$ the sense of optimality is defined by a positive cone $C(t)$ in R^p (where $C(t)$ might be defined differently for each t, but usually $C(t) = C$ as in (4a)). Thus, we can define Pareto-optimality of outcomes $y(t)$ for all $t = t_0, \dots t_f$; but what is the actual meaning of such a definition?

In this case, actual decision outcomes become *outcome trajectories* $y = \{y(t_o), \dots y(t_f)\} \in R^{pN}$ where $N = t_f - t_0 + 1$ and the sense of optimization in R^{pN} is defined by a positive cone:

$$\mathcal{C} = \prod_{t=t_0}^{t_f} C(t) \tag{15b}$$

If $C(t)$ and the sense of optimality changes considerably over time, then the ***multi-objective trajectory optimization*** is in fact only an optimization with an outcome space of much larger dimensionality. However, the interesting case is when $C(t) = C$ is independent of time (say, if we maximize a component $y_i(t)$ at t_f, we want to maximize it also at all $t = t_0, \dots t_f$) and a uniform treatment of an entire outcome trajectory is substantiated.

The essential advantage of dealing with entire trajectories of outcomes is that human mind can interpret and evaluate them holistically, by "Gestalt" — which makes it much easier to specify, for example, reference or aspiration trajectories for selected objective components. While we have difficulties when dealing in our minds with more than seven to nine objects at once, this does not mean that these objects must be necessarily single numbers; they might as well correspond to any familiar concept, shape or profile — or trajectory — that, when treated analytically, might be even described as an element of an infinite-dimensional space, but in our minds aggregates information that in a computer would be represented by a very large number of similar data.

Therefore, it is advisable to group into outcome trajectories (over time) or profiles (over space or other variables) any larger number of outcomes of similar nature, not necessarily in dynamic models. A profile or distribution of similar outcomes either in design of mechanical structures — e.g. the distribution of stress along a span of a bridge — or in business management — e.g. the distribution of profits among various branches of a company — can be as well usefully interpreted as an outcome trajectory.

If $C(t) = C = R^p_+$, then it is easy to generalize the achievement scalarizing function (13a) to the case of multi-objective trajectory optimization:

$$s(y, \bar{y}) = \min_{1 \le i \le p} \min_{t_0 \le t \le t_f} (y_i - \bar{y}_i) + \varepsilon \sum_{i=1}^{p} \sum_{t=t_0}^{t_f} (y_i - \bar{y}_i) \tag{16}$$

However, the maximization of (16) is relatively simple only in the case of linear state equations with linear constraints and outcome functions; in other cases, we must employ nondifferentiable methods for dynamic optimization.

We can also use other definition of positive cones, defining differently the multi-objective optimality of a trajectory, as well as special approaches to the normalization of the space of outcome trajectories, see (Rogowski, 1989, Makowski and Sosnowski, 1989). Relations between multi-objective trajectory optimization and multi-objective dynamic programming can be also investigated.

It should be stressed that the concept of multi-objective trajectory optimization for models with continuous time (or models described by partial differential equations in the case of profiles over space variables) can be interpreted as application of the results of the old line of research in infinite-dimensional multi-objective optimization — starting with Kuhn and Tucker (1950) and continued, between others, by Rolewicz (1975), Wierzbicki (1977), Tanino et al. (1980, 1982), Jahn (1983, 1984).

7 Dynamic aspects of uncertainty and learning

Models of uncertainty — in their various forms — are often connected with multi-objective optimization and have diverse dynamic aspects. The traditional, probabilistic form of uncertainty models has led to problems of stochastic optimization; but the concepts of sequential stochastic optimization together with related concepts such as stochastic optimization with recourse are, in their essence, both multi-objective and dynamic, see e.g. (Wets, 1983, Ermolev et al., 1987). The related solution techniques of various types — either stochastic quasi gradient approaches or large scale programming decomposition or other methods — are usually closely related to both dynamic and multi-objective optimization, see e.g. (Ruszczynski, 1989).

The same can be observed when analyzing other forms of uncertainty models. Fuzzy set models of uncertainty have been developed in close relation to multi-objective decision making and optimization, see e.g. (Seo et al., 1988). Set valued models of uncertainty have been essentially connected with dynamic and multi objective optimization — see (Kurzhanski, 1986). A general challenge in the related solution techniques — be it for probabilistic, fuzzy set or set-valued models of uncertainty — is to develop sufficiently powerful computational implementations to be applicable in repetitive, parameterized optimization in decision support systems.

The issue of learning in a decision process has been long ago perceived as essentially related to that of uncertainty — it is actually impossible to learn without errors or at least noise, see (Feldbaum, 1962). More recent studies of the mechanisms of expert decision making underline the qualitative change of decisions as the basic result of learning: experts make decisions intuitively, brilliantly but inconsistently, see (Dreyfus, 1984). An essential point is, however, that even an expert when faced with a novel problem ceases to decide fully intuitively and starts to analyze the decision situation; when a novice learns to become an expert, the role of analytical approaches to decision making is even more important.

However, in order to include the dynamic of learning in analytical approaches to decision making and support, further mathematical formalization (or idealization and abstraction) of the problem of learning and expert decision making is necessary. With a few exceptions — see e.g. (Michalevich, 1986) — there is too little attention paid to this important problem.

8 Conclusions

The main theme of this essay — that the decision analysis and decision optimization must be, in its essence, multi-objective and dynamic — has been perceived by many researchers before. The essential difference today is that we can make this idea operational — by utilizing advances in computer technology, computer programming and the contemporary achievements of various branches of mathematical programming.

References

Benson, H. P. and T. L. Morin (1977). The vector maximization problem: proper efficiency and stability. *SIAM Journal of Applied Mathematics*, Vol. 32, pp. 64–72.

Chang, S. L. (1966). General theory of optimal processes. *SIAM J. Control*, Vol. 4, No. 1.

Charnes, A., and W. W. Cooper (1977). Goal programming and multiple objective optimization. *J. Oper. Res. Soc.*, Vol. 1, pp. 39–54.

Clarke, F. (1983). Optimization and Nonsmooth Analysis. J.Wiley, New York.

Dreyfus, S. (1984) Beyond rationality. In Grauer, M., Thompson, M., Wierzbicki A. P., eds: Plural Rationality and Interactive Decision Processes, Proceedings, Sopron 1984. Springer Verlag, Berlin-Heidelberg.

Ermolev, Yu. M., and R. J. B. Wets (1987). Numerical Methods in Stochastic Programming. Springer Verlag, Berlin-Heidelberg.

Feldbaum, A. A. (1962). Foundations of the Theory of Optimal Control Systems (in Russian: Osnovy teorii optimalnych avtomaticheskikh sistem). Nauka, Moscow.

Geoffrion, A. M. (1968). Proper efficiency and the theory of vector optimization. *Journal of Mathematical Analysis and Applications*, Vol. 22, pp. 618–630.

Geoffrion, A. M. (1971). Duality in nonlinear programming: a simplified applications-oriented development. *SIAM Review*, Vol. 13, pp. 1–37.

Haimes, Y. Y. and W. A. Hall (1974). Multiobjectives in water resource systems analysis: the surrogate trade-off method. *Water Resources Research*, Vol. 10, pp. 615–624.

Henig, M. I. (1982). Proper efficiency with respect to cones. *Journal of Optimization Theory and Applications*, Vol. 36, pp. 387–407.

Ignizio, J. P. (1978). Goal programming — a tool for multiobjective analysis. *Journal for Operational Research*, Vol. 29, pp. 1109–1119.

Jahn, J. (1983). Duality in vector optimization. *Math. Programming*, Vol. 25, pp. 343–353.

Jahn, J. (1984). Scalarization in vector optimization. *Math. Programming*, Vol. 26, pp. 203–218.

Kiwiel, K. C. (1985). Methods of Descent for Nondifferentiable Optimization. Springer Verlag, Berlin.

Kreglewski, T., J. Paczynski, J. Granat and A. P. Wierzbicki (1989). IAC-DIDAS-N — a dynamic interactive decision analysis and support system for multi-criteria analysis of nonlinear models. In: A. Lewandowski and A. P. Wierzbicki, eds, Aspiration Based Decision Support Systems, Springer-Verlag, Berlin.

Kaczmarek, Z. and A. P. Wierzbicki, eds (1989). Towards Modern Dimensions of Poland — Development Dilemmas on the Verge of XXI Century (in Polish: O nowoczesny wymiar Polski — dylematy rozwoju na progu XXI wieku). Ossolineum, Warsaw-Wroclaw.

Kuhn, H. W. and A. W. Tucker (1950). Nonlinear Programming. In: Proceedings of Second Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, Berkeley, California, pp. 481–492.

Kurzhanski, A. B. (1986). Inverse problems in multiobjective dynamic optimization. In: Y. Sawaragi, K. Inoue and H. Nakayama, eds.: Towards Interactive and Intelligent Decision Support Systems, Springer-Verlag, Berlin-Heidelberg.

Li, D., and Y. Y. Haimes (1987). The envelope approach for multiobjective optimization problems. *IEEE-SMC*, Vol. 17, pp. 1026–1038.

Makowski, M., and J. S. Sosnowski (1989). Mathematical Programming Package HYBRID. In: A. Lewandowski and A. P. Wierzbicki, eds.: Aspiration Based Decision Support Systems, Springer Verlag, Berlin-Heidelberg.

Rolewicz, S. (1975). On a norm scalarization in infinite dimensional Banach spaces. *Control and Cybernetics*, Vol. 4, pp. 85–89.

Rogowski, T. (1989). Dynamic aspects of multiobjective trajectory optimization in decision support systems. In: A. Lewandowski and A. P. Wierzbicki, eds.: Aspiration Based Decision Support Systems, Springer Verlag, Berlin-Heidelberg.

Ruszczynski, A. (1989). Modern techniques for linear dynamic and stochastic programs. In: A. Lewandowski and A. P. Wierzbicki, eds.: Aspiration Based Decision Support Systems, Springer Verlag, Berlin-Heidelberg.

Salukvadze, M. L. (1971). On the optimization of vector functionals, 1. Programming optimal trajectories, 2. Analytic construction of optimal controllers (in Russian). *Automation and Remote Control*, No. 7, 8.

Salukvadze, M. L. (1979). Vector-valued optimization problems in control theory. Academic Press, New York.

Sawaragi, Y., H. Nakayama and T. Tanino (1985). Theory of Multiobjective Optimization. Academic Press, Orlando Fl.

Seo, F. and M. Sakawa (1988). Multiple Criteria Decision Analysis in Regional Planning: Concepts, Methods and Applications. D. Reidel Publishing Company, Dordrecht.

Simon, H. (1958). Administrative Behavior. McMillan, New York.

Tanino, T. and Y. Sawaragi (1980). Conjugate maps and duality in multiobjective optimization. *JOTA*, Vol. 31, pp. 473–499.

Tanino, T. (1982). Saddle points and duality in multi-objective programming. *Int. J. Systems Sci.*, Vol. 13, pp. 323-335.

Wets, R. J. B. (1983). Stochastic programming: solution schemes and approximation techniques. In: A. Bachem et al., eds.: Mathematical Programming: The State of the Art. Springer Verlag, Berlin-Heidelberg.

Wierzbicki, A. P. (1975). Penalty methods in solving optimization problems with vector performance criteria. Proceedings of the VI-th IFAC World Congress, Cambridge, Massachusetts.

Wierzbicki, A. P. (1977). Basic properties of scalarizing functionals for multiobjective optimization. *Mathematische Operationsforschung und Statistik, s. Optimization*, Vol. 8, pp. 55–60.

Wierzbicki, A. P. (1984). Models and Sensitivity of Control Systems. Elsevier, Amsterdam.

Wierzbicki, A. P. (1986). On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR-Spektrum*, Vol. 8, pp. 73–87.

Wierzbicki, A. P. (1990). Multiple criteria solutions in noncooperative game theory. Part III: Theoretical foundations. Kyoto Institute of Economic Research, Discussion Paper No. 288.

Volkovich, V. L. (1969). Multicriteria problems and methods of their solutions (in Ukrainian). In: Complex Control Problems, Izd. Naukova Dumka, Kiev.

Yu, P. L. (1972). Cone convexity, cone extreme points and nondominated solutions in decision problems with multiobjectives. WP 72–02, Center for Systems Science, University of Rochester, Rochester, New York.

Yu, P. L. and G. Leitmann (1974). Compromise solutions, domination structures and Salukvadze's solution. *JOTA*, Vol. 13, pp. 362–378.

Zeleny, M. (1973). Compromise programming. In: J. L. Cochrane and M. Zeleny, eds.: Multiple Criteria Decision Making. Univ. of S. Carolina Press, Columbia, S.Carolina.

Zeleny, M. (1982). Multiple Criteria Decision Making. McGraw Hill, New York.

Part 2

Applications of Multiple Criteria Optimization

Multi-Factor Financial Planning: An Outline and Illustration

Marc Goedhart
Tinbergen Institute
and
Jaap Spronk
Dept. of Finance & Investment
Erasmus University Rotterdam
P.O. Box 1738, 3000 DR Rotterdam, the Netherlands

Abstract

This paper deals with the support of strategically oriented financial planning processes in business firms. In handling a financial planning problem, the decision maker has to deal with a number of complications. In this paper special attention is paid to the risk with regard to the outcomes of the financial plan and the existence of multiple, conflicting goals. Various participants in the firm make their demands on financial policy. As these demands may well be conflicting, the planner is faced with a multi-objective decision problem.

In financial literature, risk is usually modelled as the variance of cash flows or in terms of their covariance with the return on the financial market portfolio. Risk is then usually incorporated in the net present value of the cash flows by using a risk adjusted discount rate. The existence of multiple goals is usually abstracted from, or is considered to be included in the net present value. The only goal the financial planner needs to be concerned about is the maximization of the firms market value.

In this paper, an interactive approach to financial planning is presented. In this approach, risk is modelled by means of so-called multi-factor risk models. Multiple goals are explicitly accounted for in our model through the use of an interactive goal programming method. The use of the interactive approach will be numerically demonstrated by means of an exemplary planning problem. In this problem, a selection is to be made from a set of capital investment projects while taking account of the financing consequences over time.

1 Introduction

In this paper, financial planning is seen as a structured process of identification and selection of present and future capital investment projects (including disinvestments) while taking account of the financing of these projects over time. This process can be visualised as follows:

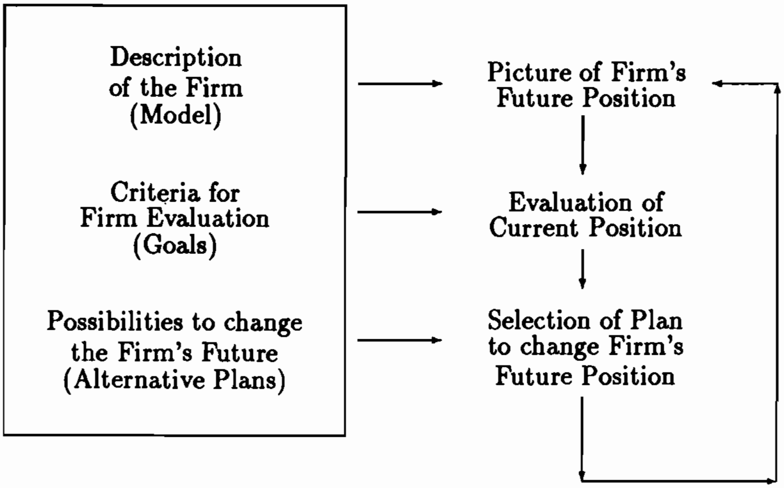


Figure 1: Financial Planning Process

The firm has to derive its right to exist from the fact that it creates a surplus of value to society. In other words,the sum of contributions from the firm's participants should at least equal and preferably be outweighed by the total of effects the firm has in return on these participants. Thus the firm combines inputs such as labour, money, goods and information into direct outputs such as goods and services. The output of the firm also includes indirect effects which may vary from positive effects like job satisfaction to negative effects like pollution. In this view, the firm is continuously managing a set of exchange relations with its participants. Clearly, there are different types of exchange relations. For instance, a relation can be a long-lasting one (e.g. with a part of the firm's employees) or be more volatile (e.g. a deal in a fully competitive market). Relations may also differ in the sanctionary power of the parties involved, which may vary from a client's power to refuse the purchase of the firm's products to the power to strike (employees) or to raid the firm and to strip its assets.

The kind of exchange relations a particular firm has to deal with depends on the type of its organisation. As compared with market deals, long lasting relations have both merits (e.g. a good possibility to combine the expertise of different participants, lower search and contracting costs, lower uncertainty through information exchange) and costs (e.g. bureaucracy, influence on the incentive structure). The size of both merits and costs beyond doubt depends on the political-cultural environment of the firm.

Obviously, the firm's exchange relations are dynamic: the firm can to a large extent choose its own participants who, in turn, generally have their alternatives outside the firm. This implies that the resultant effect of these relations is dynamic as well. In other words: the firm has to deal with a dynamic goal complex, a multiplicity of goals which vary over time. In our opinion, any decision supportive approach to financial planning should take account of the reasonably well established fact that people are not very good in assessing reliable probabilities to the outcomes of future events. This may be partly explained by pointing at the limitations of the human mind. Another reason is of an even more fundamental nature: many future outcomes are contingent on decisions which are still to be made. In the case of capital investment projects, for instance, the future cash flows can generally not be well represented by some probability distribution alone. In addition it is often necessary to describe as well all the 'rights' and 'duties' connected with the project. For instance, the right to expand or to abandon the project or some legal duty to keep people on the payroll even in case it would be more economical to fire them. One might consider the use of decision trees to describe this type of situations. But in most practical cases this is impossible because of the number of possible outcomes being too large and/or because the timing of the potential outcomes (when something will happen) is largely unknown. The picture becomes even more complicated if one realises that the outcomes of a project are often not only contingent on the future decisions of the decision maker himself, but may also be contingent on the future decisions of the firm's participants, which are in their turn contingent on the firm's decisions. In other words, the way the firm manages its exchange relations influences the uncertainty surrounding the firm's cash flows. From the above observations it is clear that the alternative plans to be considered by the decision maker are hard if not impossible to evaluate on the exclusive basis of the objective adopted by financial theory: the maximization of the share holders wealth. One problem is the contingencies, another is the game like nature of at least part of the firm's dynamic goal complex. However, cash flow remains one of the central concepts in developing a financial plan. The owners of the firm will evaluate the firm in terms of alternative investment opportunities. An obvious possibility is to invest in the stock market. This gives a clear reason to use the firm's market value as a benchmark for evaluating the firm and its investments. And to close the circle: the firm's market value directly depends on the firm's cash flows and the associated risk. If the firm's cash flows decrease and/or its riskiness increases, its market value will drop. As such, the shareholders may de facto be the most powerful group of participants of the firm. However, it is equally clear that the firm should not neglect the other participant's interests, both for its own sake and in the interest of the stockholders. In summary, the firm should for each financial plan try to assess all effects considered to be important by the most influential participants, taking account of the most important contingincies involved.

In this paper we describe a framework for financial planning in the firm which is based on two important developments in financial theory:

a. the use of multi-factor models to describe probability distributions of returns; and

b. the revival of the concept of flexibility and its valuation through option pricing theory and contingent claims analysis.

We will show how these two ingredients can be used in a multiple criteria approach to financial planning in which the decision maker can systematically investigate the set of alternative financial plans, taking account of the most important parties, the most important uncertain factors and the most important contingincies, rights and duties involved. Preliminary experiences have given us the conviction that the framework has a good potential to be understood and used in practice.

The general ideas behind the proposed framework are outlined in the next section. In section 3 the framework itself is described after which in section 4 some elements of the framework are illustrated by means of a practical example. Section 5 is devoted to some aspects of the operationalization of the framework. Our conclusions are summarized in section 6.

2 The multi-factor framework and contingent claims

In modelling uncertainty one has to find a compromise between precision on the one hand and comprehensibility and managability by the model user on the other hand. Very often, uncertain outcomes of decision alternatives (projects, plans, etc.) are modelled as probability distributions defined on the range of possible outcomes. Decision makers are then required to assess the value of some parameters (e.g. mean and variance) of these distributions. In addition, decision makers often have to express their preferences (e.g. utility values) with respect to the uncertain outcomes. The thusly provided information is then used in the formulation of an objective function (e.g. the expected utility of the decision alternatives) or, alternatively, in formulating chance constraints. For the reasons given above we deviate from this approach by assuming that the uncertain phenomena met in financial planning can generally not be modelled precisely. However, following (Hallerbach and Spronk, 1986), we assume that for many of these phenomena at least some structure can be found by using multi-factor models. The results of a financial plan (to facilitate the exposition in this section, we take the firm's cash flow as the only relevant outcome variable) will depend on the one hand on the decisions made by the firm and on the other hand on the various forces and influences from its dynamic environment. We assume that, in general, it is very hard if not impossible to define a complete probability distribution over the value of the cash flows which may result. Instead, we assume that the firm is able to define its expectations concerning these cash flows and, in addition, that it is able to assess the sensitivity of these cash flows for unexpected changes in a number of factors which influence these cash flows. Consequently, the effect of a decision can be modelled as an expected level of the cash flows plus a series of sensitivities for unexpected changes in a number of factors influencing these cash flows. The firm does not necessarily know how these factors themselves will change in the future. On basis of this way of modelling, a firm can estimate the firm's aggregate sensitivity (i.e. the sensitivity of all decisions combined) for the various factors it has found to be important. In this way, a given financial plan of the firm can be evaluated in terms of an expected cash flow development accompanied by a vector of sensitivities describing what deviations from this expectation could arise if one or more of the identified factors would get another value than expected. Clearly, the aggregate sensitivities can be treated by the decision makers as goal variables, which can be traded off against each other and against other goal variables, possibly within the framework of a formal multi-criteria analysis (see section 4). Such an analysis might show for instance that the firm's sensitivity for unexpected wage rate changes can be lowered, but at the price of increasing the firm's sensitivity for unexpected changes in the dollar rate. It is conceivable that a firm is also evaluated in other terms than cash flows alone (e.g. employment level, labour relations, market power, etc.). Such a broadened evaluation can be done in a straightforward manner, be it that the amount of calculations increases and that one may want to investigate the interrelations between the evaluation criteria. Many different factors will influence the future cash flows of the firm. Ideally, all these factors could be identified and all corresponding sensitivities could be measured. Thus, a fully specified multi-factor risk model could be generated. In practice, however, several problems with regard to the assessment of the factors and factor sensitivities will occur.

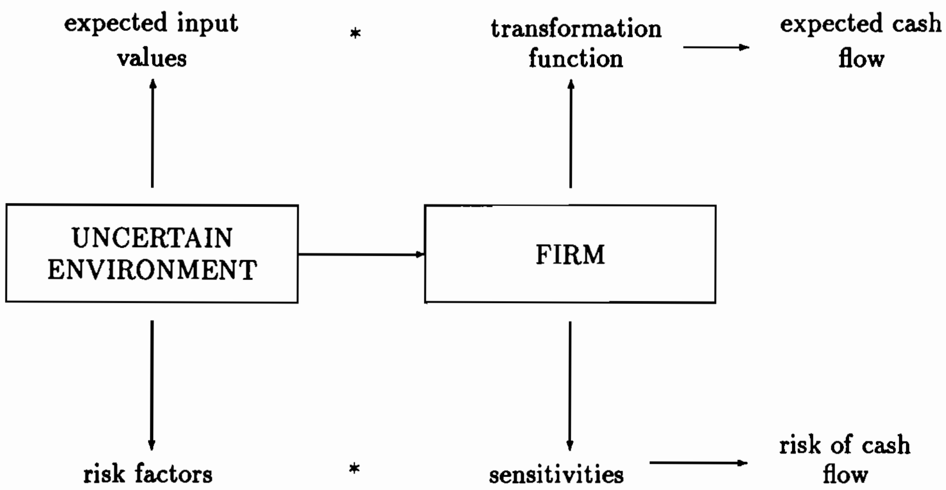


Figure 2: Multi-Factor Risk Modelling

There will be factors which are well-identifiable but less well measurable, for example because of limitations in the data or because the factor concerned can not easily be operationalized. In addition, factors exist which are not well identifiable because of their too infrequent occurrence. In general, some of the firm's risk factors will be well-identifiable and well-measurable whereas others will be more difficult or even too difficult to identify or to measure. Apart from difficulties concerning the measurement of risk factors itself, problems can arise in measuring the cash flow's sensitivities for these factors. Thus, when modelling the influence of the uncertain environment on the firm's cash flow, two kinds of complications can occur. Firstly, the modelling of the environment itself by risk factors may be troublesome, and secondly, the modelling of the influence of the environment on the cash flows by means of sensitivities may be arduous (see figure 2).

The framework discussed here stresses the sensitivity of the cash flows for unexpected changes in well-measurable factors (see figure 3). However, in the evaluation of a plan, attention should also be paid to the identifiable but less well-measurable factors hidden in the specific risk component. The least to be done is to develop a list of this type of factors which can be used as an additional check-list for the evaluation of the plans. With respect to the factors which are less well identifiable one cannot do much more than to develop the reserves and the ability to handle unforeseen situations. After the identification of the most important factors and the assessment of the firm's sensitivity for these factors, i.e. the actual multi-factor risk modelling process, a number of questions remains.

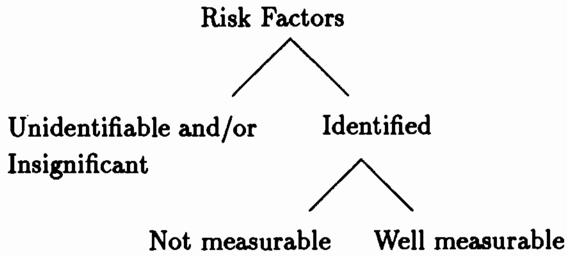


Figure 3: Classification of Risk Factors

These questions concern the choice of the appropriate actions to be taken by the firm when faced with a particular configuration of factor sensitivities. A first question is whether and, if so, how the firm is willing to accept certain exposures to unexpected changes in the factors. Basically, there are two ways to change the firms aggregate factor sensitivities. The first way is to use contingent instruments such as investment or financing projects that create certain rights or duties, the second is the employment of non-contingent measures.

Non-contingent actions resemble 'traditional' hedging strategies such as in, for example, an investment portfolio of securities. For a planning problem in a firm such a strategy implies that an unacceptable pattern of factor sensitivities is limited or even neutralized by an investment in a project that generates a series of factor sensitivities of opposite sign. In this manner, both favourable and unfavorable effects of unexpected factor changes on the firm's cash flow can be reduced or perhaps even eliminated. An example of such a strategy of non-contingent hedging is the acquisition of energy plants by a transportation company. The adverse effects of for example oil price increases will be (partly) compensated for by opposite effects of the price rise in the energy producing business, and vice versa.

Contingent instruments give the firm the possibility to eliminate or reduce only the adverse effects of factor changes, and to profit from favourable effects. We distinguish between two kinds of flexibility that may be created by means of contingent instruments: directed flexibility, which has the character of an insurance against some specific risk, and on the other hand undirected flexibility, which will be also referred to as 'elbow-room'. One possibility to neutralize or to limit a risk beforehand is by 'buying an insurance' with respect to this risk (where 'buying an insurance' should be understood in a broad sense). For instance, firms often insure themselves against the negative consequences of unexpected changes in factors such as fire or exchange rate fluctuations (the latter can often be insured by buying valuta options). On the other hand, the firm can assure itself of the positive effects of unexpected changes in the factors (e.g. by acquiring the exclusive selling rights of a product in development). For this kind of risk handling to be appropriate, it is necessary that management can at least identify the risk factors it wants to be protected from. Another possibility to face risks, instead of buying an insurance, is to create sufficient elbow-room in the firm to be able to react adequately to an unexpected change in some possibly not yet identified factor if and at the moment it occurs. An example is not to insure against fire but instead to make a large enough reservation to be able to bear the negative consequences of fire. The importance of some elbow-room is illustrated by a comparison of the histories of PanAm and Braniff (example from Casey and Bartczak, 1985). At a certain moment both firms generated a negative net cash flow. Braniff went bankrupt in 1982 while PanAm could survive longer by selling two important assets (Intercontinental Hotel and PanAm building). An evaluation of PanAm based on a cash flow analysis alone — i.e. without taking account of its 'elbow-room' — would clearly have given a wrong answer. A firm having a good and stable policy for recruitment and selection of personnel will produce less 'surprises' than a firm with a clumsy ad hoc policy. Of course, an evaluation backing an important decision will pay attention to such a factor, even if it is less well measurable. Creating elbow-room can be viewed as buying an option. If an unexpected change of a factor materializes, the option holder has the right to use the elbow-room to react to this change. The possessor of the elbow-room is of course obliged to bear the consequences if the elbow-room is not sufficient. Elbow-room is often labelled as flexibility[1], where a distinction is made between operational and financial flexibility (cf. Kemna and Van Vliet, 1984, and Kemna, 1988). Examples of operational flexibility are the possibility to quickly adapt production (e.g. with respect to production volume or to product specification) according to the changing needs of the product's consumers. Already twenty years ago, the importance of financial flexibility was stressed by Donaldson, (1969, 1984), and in the Netherlands by Diepenhorst, (1962) (the latter made the useful distinction between defensive and offensive flexibility). Examples of financial flexibility are unused reserves (cash surpluses, unused credit facilities), unused debt capacity, the capability to reduce expenditures and the earlier mentioned possibility to sell assets. Not surprisingly, the creation and maintenance of elbow-room is not without costs: flexibility has its price (cf. Kemna and Van Vliet, op cit.). In efficient markets we would expect risk handling by creating flexibility to be more expensive than by using non-contingent instruments. For example, in financial markets, we would expect that hedging a currency exposure by means of currency option requires a higher investment than hedging the same position by means of a forward contract. The option strategy gives the buyer the *right* to exercise his currency option in case of unfavorable currency rate changes, while still enabling him to profit from favourable changes. The forward transaction, however, hedges the currency position against both favourable and unfavorable changes in the currency rate. Once the transaction has been made, the holder of the currency position can no longer profit from favourable currency rate changes. For real markets, an analogous line of reasoning would be valid if those markets were efficient. As this is not always the case, risk reduction by creating flexibility need not always be more expensive.

[1] We would rather use the term "undirected flexibility" for elbow-room, as opposed to "directed flexibility", which refers to the kind of 'insurance' mentioned earlier in the text.

3 Multi-factor financial planning

In the multi-factorial approach to financial planning, the idea of replacing probability distributions by multi-factor representations is combined with the concept of contingent claims and with multiple criteria decision procedures. In broad lines, the resultant approach can be described as follows.

a. Identification of the firm's most important environmental factors. A distinction can be made between operational and financial factors. Operational factors relate to the firm's net cash flows. Examples are the price of raw materials, labour and the price of energy. Financial factors relate to the cash flows flowing from and to the suppliers of the firm's capital. Examples are interest rates and risk premiums required by the market.

b. Identification of projects, both real and financial. Real projects may vary from stopping a certain production line to the take-over of a competitor. Financial projects may vary from a modest one-year bank loan to a substantial emission of new stock.

c. Calculation of the expected cash flows of the firm as it is. (Obviously, the expected development of other output variables can be calculated as well. For ease of exposition hereafter only cash flow is assumed to be relevant). To calculate these expected cash flows, a 'base case' policy has to be defined, assuming that the environmental factors defined in (a) will adopt their expected values.

d. Estimation of the expected future cash flows of the projects. Here, the expected incremental cash flows are to be calculated. That is, the difference between the expected cash flows of the firm including the project and the expected cash flows of the firm not including the project (i.e. the expected values calculated in (c)). As in (c), the expected incremental cash flows are calculated on the assumption that the environmental factors adopt their expected values.

e. Measurement of the firm's and the projects' sensitivities for unexpected changes in the environmental factors. The effect of the expected changes in the environmental factors has already be included in the expected cash flows calculated in (c) and (d).

f. Identification, specification and, if possible, valuation of the options ('rights and duties') available to the firm as it is and of those associated with new projects. The valuation of options is a relatively new area which is producing a stream of exciting results. Notwithstanding the large number of successes in that field (see Kemna, 1988, for an overview) a lot of questions remain unanswered. Especially with respect to the valuation of real options one may expect more or less exact answers only in a limited amount of cases. Clearly, options may change the magnitude of the sensitivities of the cash flows. Alternatively, options may limit the effect of one or more unexpected factor movements without changing the sensitivities as such.

g. Both the firm as it is and the projects at hand are described by multidimensional profiles, consisting of the expected values calculated in (c) and (d) and of the sensitivities for unexpected changes in the environmental factors calculated in (e) and if necessary corrected for the effect of the options mentioned in (f).The sensitivities estimated in the preceding steps relate to the cash flows streaming to the suppliers of capital. Given the firm's market value as a primary objective, one should try to find out how the financial market valuates different constellations of risk factors.

h. On basis of these multidimensional profiles of the firm and the projects, the problem is to find the combination of projects which contributes best in terms of the firm's objectives: this is done by IMGP (see Spronk, 1981, 1985), in which the decision maker has the possibility to condition the set of possible project combinations by setting and systematically changing a series of goal constraints.

i. The result of (h) can be a single plan or a set of plans which is subjected to a secondary analysis. For instance, it is possible to investigate how a plan would perform given a constellation of unexpected factor changes (of course, several factor constellations can be investigated). Furthermore, one may want to evaluate the generated plans in terms of objectives which were not explicitly included in the model.

4 Illustration

In this section we will address the steps (g) to (i) as indicated in the multi-factorial approach to financial planning. These steps are concerned with the modelling of the firm and its projects in terms of expected values of cash flows and their sensitivities for changes in environmental factors, and furthermore, with the selection of a plan or set of plans and subsequent secondary analysis. In our example, the firm is represented by a financial planning model which covers ten periods. Uncertainty with regard to the firm's cash flows is modelled by means of multi-factor relations. It is assumed that two operational risk factors exist: the oil price level and the wage level, which are both represented by indices. Furthermore, a financial risk factor is embodied in the interest rate level[2] over outstanding debt. Management is to formulate a financial plan over the ten year planning period showing the investment projects undertaken and the way in which these projects will be financed over time. The instruments are the available investment projects, the dividends paid out in each period and the amount of debt attracted in each period. Equity issues, cash holdings and taxes are not considered in this example. Goal variables are the present value of the dividend stream discounted at the risk free rate, the number of dismissals during the planning period, the sensitivities of the net cash flow in each year for unexpected changes in oil prices and in the wage level, and finally, the sensitivity of the net cash flow for unexpected changes in the interest level.

4.1 Instruments

Three categories of instruments are available to the firm to manipulate its goal variables: the choice of investment projects, the amount of dividends paid out annually and the debt attracted in each year. With regard to the investment projects we assume that the

[2] We do not adjust the discounted value of the dividend stream for possible changes in the risk free rate of return.

relevant attributes are the annual expected net cash flow, the annual sensitivities for the two indicated operational risk factors and the incremental employment from the project. Incremental employment associated with an investment project can also be negative, which implies a number of dismissals. For simplicity, it is assumed that projects may also be partially adopted, which is defined by:

$$0 \leq x_j \leq 1 \qquad \text{for} \quad j = 1, \ldots, J; \tag{1}$$

where j is the project index (J=20). However, a discrete/integer approach to the project selection problem is also possible within the proposed framework. In table 1 the available investment projects and some of their effects on the model's variables are given.

	Cash Flow										EMP
Project	0	1	2	3	4	5	6	7	8	9	
0	500	550	500	450	460	420	400	480	420	430	-20
1	-100	20	15	9	12	18	15	20	10	20	-40
2	0	0	0	-30	20	12	15	15	25	25	-20
3	-200	25	20	18	16	17	22	15	17	16	-15
4	-100	15	12	11	10	15	16	18	20	20	-16
5	-500	58	50	60	50	55	62	60	55	50	-40
6	0	0	0	-2500	500	550	620	600	550	500	-40
7	0	-150	75	75	75	0	0	0	0	0	10
8	-80	40	10	15	10	20	0	0	0	0	-20
9	0	0	0	0	0	0	0	-100	50	50	-4
10	-100	30	30	30	-50	30	30	30	30	30	10
11	-20	6	7	6	7	0	0	0	0	0	0
12	0	-20	6	7	6	0	0	0	0	0	-15
13	0	0	-20	6	7	6	7	0	0	0	-20
14	0	-80	15	15	15	15	15	15	15	15	-40
15	-200	18	18	18	18	18	20	20	20	20	-15
16	-50	10	20	40	0	0	0	0	0	0	-10
17	-130	20	20	50	50	40	0	0	0	0	-30
18	0	0	0	-130	20	20	50	50	40	0	30
19	-125	15	18	16	10	12	-10	14	-10	20	10
20	-250	50	60	40	30	30	50	40	30	20	-40

Table 1: Some attributes of the available projects[3] (in \$ 1,000)

The second category of instrumental variables in the model is the amount of debt attracted in each period. To finance its capital investments, the firm can choose from a number of large financing projects such as bond issues. Thus, the financing problem is partly considered as one of selecting the best financing projects. Additional funds can be raised by attracting one-period loans, which are fully repaid at the end of the period. All funds needed for investment are raised either internally from other projects' cash flows,

[3]For additional project data, see Appendix 1

or externally from one-period loans at the prevailing interest rate r_L and bonds issued at a fixed coupon rate. The bonds pay a fixed annual coupon rate and have a maturity varying from 5 to 9 years, as shown in table 2. Bond principal is repaid at maturity. In this case we assume that the costs to a firm of financing by means of bonds are higher than the expected costs of financing by a series of short-term loans. But in comparison to the one-period loans, bonds offer the issuing firm a reduction of interest risk.

	Cash Flow								
Bond	0	1	2	3	4	5	6	7	8
0	200	16	16	16	16	16	16	16	216
1	150	11.3	11.3	11.3	11.3	161			
2	300	24	24	24	24	24	324		
3	400	32	32	32	32	32	32	32	432

Table 2: Financing possibilities by bonds (in \$ 1,000)

In our model, a substitution of one-period loans by long-term bonds would imply a reduced interest sensitivity and a lower net present value of the cash flow after interest, which is available for dividend payments (see following sections). In this way, a trade-off arises for the firm in formulating its financial policy: increases of the cash flow to equity can be realized at the cost of facing a higher sensitivity of that cash flow to unexpected changes in the future interest rate. Equity issues and lending facilities are not considered in this example.

Furthermore, the firm can influence its goal variables through the dividend policy. The amount of dividend paid out in each period is the third instrument. With this variable, Div_t, management can directly influence the first goal variable in the model. To smoothen the variability of the dividends management has defined a growth path, which is defined by an upper bound and a lower bound for the growth rate of the dividend stream:

$$Div_t \leq Div_{t-1} \cdot (1+g^+) \quad \text{for} \quad t=1,\ldots,9; \tag{2a}$$

$$Div_t \geq Div_{t-1} \cdot (1+g^-) \quad \text{for} \quad t=1,\ldots,9; \tag{2b}$$

where g^+ is the upper, and g^- the lower growth rate level.

4.2 Definitional equations

In the model, the consequences of accepting investment projects for the aggregate net cash flow are figured as follows:

$$\overline{CF}_t = \sum_{j=0}^{20} x_j \cdot \overline{CF}_{jt} \qquad \text{for} \quad t=0,\ldots,9; \tag{3}$$

$\overline{CF}_{jt}$: expected net cash flow for project in period t;

Here, x_0 represents the firm as it is with its existing assets and associated cash flows and factor exposures. Assuming that disinvestments are not possible, x_0 is fixed at 1.

The annual factor sensitivities at the overall firm level are derived from the individual project sensitivities by simple summation[4]:

$$SensOil_t = \sum_{j=0}^{20} x_j \cdot SensOil_{jt} \quad \text{for} \quad t = 0, \ldots, 9; \tag{4}$$

$$SensWage_t = \sum_{j=0}^{20} x_j \cdot SensWage_{jt} \quad \text{for} \quad t = 0, \ldots, 9; \tag{5}$$

With regard to employment in the firm, we assumed that attention is primarily directed at the number of dismissals resulting from a certain financial plan. Therefore, only negative incremental employment is considered relevant. Total incremental employment, ΔEmp, is defined as follows:

$$\Delta Emp = \sum_{j=1}^{20} x_j \cdot \Delta Emp_j; \tag{6}$$

$$\Delta Emp - Emp^+ + Emp^- = 0; \tag{7}$$

$$Emp^+, Emp^- \geq 0; \tag{8}$$

$$Dismiss = Emp^-; \tag{9}$$

The number of dismissals, *Dismiss*, is a goal variable in the model that is to be minimized.

The sources and uses of funds constraints are given by:

$$\sum_{j=1}^{20} x_j \cdot \overline{CF}_{j0} + L_0 - Div_0 + \sum_{k=1}^{K} y_k \cdot CF_{k0} = 0; \tag{10}$$

$$\sum_{j=1}^{20} x_j \cdot \overline{CF}_{jt} + L_t - L_{t-1} \cdot (1 + r_L) - Div_t + \sum_{k=1}^{K} y_k \cdot CF_{kt} = 0; \tag{11}$$

$$\text{for} \quad t = 1, \ldots, 8;$$

$$\sum_{j=1}^{20} x_j \cdot \overline{CF}_{j9} - L_8 \cdot (1 + r_L) - Div_9 + \sum_{k=1}^{K} y_k \cdot CF_{k9} = 0; \tag{12}$$

The effects of bond issues are modelled with the use of the bond variables y_k and the corresponding — certain — cash flow CF_{kt} created by bond issue k in year t. L_t represents the amount of one-period loans attracted in period t, which is repaid with interest at the expected rate r_L[5] in the following period. In this example, no loans are outstanding at the beginning of the first period or at the end of the last period. In each year the sensitivity of the cash flow to equity for unexpected changes in the debt rate is simply the amount of one-period loans attracted in the previous year:

$$SensInt_t = 0.01 \cdot L_t \qquad \text{for} \quad t = 0, \ldots, 8; \tag{13}$$

[4]Cash flow sensitivities are defined as the derivative of the stochastic project cash flow with regard to the factor index; e.g.:

$$SensOil_{jt} = \frac{dCF_{jt}}{dOil_t} \tag{4a}$$

[5]By assuming a certain term structure of interest rates, one should also allow for expected short-term interest rates to vary over time.

Thus, *SensInt* gives the absolute change in the cash flow to equity for a percentage point change in the interest rate due over the outstanding debt. Clearly, interest sensitivity is always positive in our model as we abstracted from the possibility of lending funds.

4.3 Goal variables

As argued in section one, management finds itself confronted with a dynamic goal complex. In this example, the firm has to make a selection from the set of possible investment and financing decisions which renders the best contribution to the objectives. By means of IMGP the selection is made by a systematical reduction of the solution space as conditioned by a series of goal constraints. Here, the dynamic goal complex is represented by the following goal constraints:

1. Present value of the dividends;
2. Number of dismissals;
3. Maximum positive sensitivity of the net cash flow to unexpected changes in the oil price index;
4. Maximum negative sensitivity of the net cash flow to unexpected changes in the oil price index;
5. Average sensitivity of the net cash flow for unexpected changes in the wage rate;
6. Average sensitivity of the yearly dividends for unexpected changes in the debt rate.

The present value of the dividend stream represents the link with the capital market, or more specifically, with the shareholders. As the shareholders form an influential group of participants, this may be a very important goal variable. In discounting future dividends, the risk free rate of return is used instead of a risk adjusted rate. The present value as calculated in our model only incorporates the time value of money and not a risk premium. Risk is modelled by means of sensitivities for the three indicated risk factors. Therefore, the present value of the dividends is defined as:

$$PvDiv = \sum_{t=0}^{9} \frac{Div_t}{(1+r_f)^t}; \tag{14}$$

where r_f is the risk free rate of return and $PvDiv$ is to be maximized.

The second goal variable in the model concerns the level of employment in the firm. As the labour force also constitutes an important group of participants, management will consider the consequences of its capital investment plan also in terms of employment conditions. We assume that management wants to avoid struggles with its employees and that therefore the number of dismissals during the planning period is to be minimized:

$$Dismiss = Emp^{-}; \tag{15}$$

Our first goal variable explicitly excluded a risk adjusted discount rate. In this model, risky cash flows from investment projects or financing arrangements are modelled in multi-factor relations[6]. The sensitivities from these multi-factor relations lead to the resulting

[6]The approach conceptually resembles a certainty equivalent method for evaluating risky cash flows.

four goal variables. Three risk factors are thought to influence the firm's cash flows: energy prices, as pictured by an oil price index, wage rates and the interest rate over the firm's debt. As the sensitivities for the first two operational risk factors vary over both projects and years, a method had to be found to summarize these figures. For the oil price sensitivities a minimax approach combined with a maximin approach was chosen. The minimax approach minimizes the maximal sensitivity for oil price changes over the planning period. Thus, it gives management an impression of the largest possible positive exposure of the net cash flow to oil price movements over the forthcoming ten years for a certain investment plan. In mathematical terms:

$$SensOil^{+} \geq SensOil_t \quad \text{for} \quad t = 0, \ldots, 9; \tag{16}$$

$$SensOil_t = \sum_{j=0}^{20} x_j \cdot SensOil_{jt} \quad \text{for} \quad t = 0, \ldots, 9; \tag{17}$$

and $SensOil^{+}$ is to be minimized. An analogous method is used with regard to the negative exposure of the cash flow to oil price movements. Now, the minimal sensitivity over the planning period is maximized, or equivalently, the maximal negative sensitivity is minimized:

$$SensOil^{-} \leq SensOil_t \quad \text{for} \quad t = 0, \ldots, 9; \tag{18}$$

We assumed that wage rate changes are always negatively correlated with the net cash flow in the firm: wage rate increases invariably reduce the net cash flow. Therefore, we could suffice with a minimax approach for the handling of wage rate exposures. For illustrational purposes, another method of summarizing the risk associated with wage rate movements was chosen. The average sensitivity for wage rate movements over the ten year planning period constitutes the fifth goal variable, which is to be minimized:

$$WageSens_t = \sum_{j=0}^{20} x_j \cdot WageSens_{jt} \qquad \text{for} \quad t = 0, \ldots, 9; \tag{19}$$

$$\overline{WageSens} = \sum_{t=0}^{9} 0.10 \cdot WageSens_t; \tag{20}$$

Thus far, two operational risk factors affecting the firm's net cash flow before interest payments were considered. As an example of a financial risk factor the interest rate over the outstanding debt was chosen. It was previously argued that the value for the debt rate sensitivity was given by the amount of debt attracted in the former period. Because interest exposure always has a positive value, the last goal variable is the average interest sensitivity over the planning period, which is to be minimized:

$$\overline{IntSens} = \sum_{t=0}^{8} \frac{1}{9} \cdot IntSens_t; \tag{21}$$

4.4 Selection of a financial plan

As mentioned earlier, Interactive Multiple Goal Programming (IMGP) is used here to obtain the combination of investment projects, bond issues, annual loans and dividends which best contributes to the firm's objectives. Basically, objectives are modelled as goal constraints in IMGP. By manipulating these goal constraints, the decision maker can systematically condition the set of possible values for the instrument variables. The manipulation of goal constraints takes place in a series of iterations in each of which the decision maker is confronted with a so-called potency matrix. In this matrix, for each goal variable a feasible range is presented defined by an "ideal" and "pessimistic" value. The ideal value for the goal variable in question is the optimum for that individual variable taking the pessimistic values for the other goal variables as constraints to the optimization. The pessimistic values are provided by the decision maker and represent minimum[7] values acceptable. By successively raising these pessimistic values, the decision maker can reach a final solution, or set of solutions, — this depends on the extent to which he conditions the solution space. In each iteration, the decision maker can assess the consequences of a proposed improvement of a certain pessimistic value in terms of reduced ideal values for the remaining goal variables in the next potency matrix. If the costs of the improvement are too high according to the decision maker, the improvement can be reverted. Thus, the trade-offs between various goal variables are revealed.

For the case at hand, the first potency matrix of the first solution is given in table 3. The range over which a final solution may be chosen is quite large for some of the goal variables. This holds especially for the number of dismissals and the maximum sensitivity for changes in the oil prices.

If desired, the decision maker could even reduce the maximum oil sensitivity to such an extent that it would be negative in each future period. Some other results are shown in table 4 which gives the set of investment projects selected when each of the goal variables is optimized independently to its ideal value in the potency matrix. The table shows that the available projects differ substantially in terms of their contributions to the optimization of the goal variables. For different goal variables, different projects are selected.

Goal	Value	Iteration Number				
Variable		1	2	3	4	5
Pv-Div	Ideal	3,555.493	3,548.070	3,509.402	3,494.861	3,475.443
	Pessimistic	3,163.327	3,163.327	3,163.327	3,163.327	3,163.327
Dismissals	Ideal	0	0	0	0	0
	Pessimistic	245	245	245	75	50
SensOil-	Ideal	-8.000	-8.000	-10.777	-13.888	-19.490
	Pessimistic	-143.800	-50.000	-50.000	-50.000	-50.000
SensOil+	Ideal	-109.800	-20.000	-20.000	-20.000	-20.000
	Pessimistic	47.000	47.000	47.000	47.000	47.000
SensWage	Ideal	1.542	1.542	1.542	1.940	2.586
	Pessimistic	75.500	75.500	75.500	75.000	75.000
SensInt	Ideal	0.000	0.000	0.000	0.000	0.000
	Pessimistic	134.900	134.900	50.000	50.000	50.000

[7] For goal variables to be minimized, the pessimistic values would of course express acceptable maximal values.

Goal Variable	Value	Iteration Number 6	7	8	9	10
Pv-Div	Ideal	3,475.443	3,475.443	3,447.569	3,446.508	3,403.397
	Pessimistic	3,375.000	3,375.000	3,375.000	3,375.000	3,375.000
Dismissals	Ideal	0	0	0	3	20
	Pessimistic	50	50	50	50	50
SensOil-	Ideal	-19.505	-26.500	-27.526	-27.526	-27.526
	Pessimistic	-50.000	-50.000	-50.000	-50.000	-35.000
SensOil+	Ideal	0.000	0.000	0.000	0.000	0.000
	Pessimistic	47.000	10.000	10.000	10.000	10.000
SensWage	Ideal	3.189	3.189	3.189	3.189	6.545
	Pessimistic	75.000	75.000	75.000	20.000	20.000
SensInt	Ideal	5.700	5.700	5.700	5.700	7.580
	Pessimistic	50.000	50.000	25.000	25.000	25.000

Goal Variable	Value	Iteration Number 11	12	13	14	15
Pv-Div	Ideal	3,387.696	3,380.343	3,380.343	3,380.343	3,380.343
	Pessimistic	3,375.000	3,375.000	3,375.000	3,375.000	3,375.000
Dismissals	Ideal	20	24	24	24	24
	Pessimistic	30	30	30	30	30
SensOil-	Ideal	-31.073	-33.347	-33.347	-33.347	-34.029
	Pessimistic	-35.000	-35.000	-35.000	-35.000	-35.000
SensOil+	Ideal	0.000	0.000	0.000	0.000	0.000
	Pessimistic	10.000	10.000	5.000	2.500	1.000
SensWage	Ideal	14.872	15.637	15.637	15.512	15.698
	Pessimistic	20.000	20.000	20.000	20.000	20.000
SensInt	Ideal	16.410	16.410	16.410	16.410	16.410
	Pessimistic	25.000	20.000	20.000	20.000	20.000

Goal Variable	Value	Iteration Number 16	17	18
Pv-Div	Ideal	3,380.343	3,376.667	3,375.081
	Pessimistic	3,375.000	3,375.000	3,375.000
Dismissals	Ideal	26	28	28
	Pessimistic	30	30	28
SensOil-	Ideal	-34.121	-34.484	-34.975
	Pessimistic	-35.000	-35.000	-35.000
SensOil+	Ideal	0.000	0.000	0.085
	Pessimistic	1.000	1.000	1.000
SensWage	Ideal	15.698	15.852	17.271
	Pessimistic	17.500	17.500	17.500
SensInt	Ideal	16.410	16.410	17.440
	Pessimistic	20.000	17.500	17.500

Table 3: Potency matrices during iterations

	Goal Variable					
Project	Pv-Div	Dismiss	SensOil-	SensOil+	SensWage	SensInt
0	1	1	1	1	1	1
1	1			1	1	
2	1			1		
3				1		
4				1		
5		0.625	1			
6	1		1		0.05	
7	1	1				0.07
8	1			1	1	
9			1			0.36
10	1	1	1		1	
11	1					
12	1		0.48		1	
13	1			1	0.79	1
14	1		1		0.38	
15		1		0.36		
16	1			1	1	
17	1		0.88		1	0.19
18	1	1		1		0.22
19		1	1		0.45	
20	1			1	1	

Table 4: Investment projects selected when different goal variables are optimized.

The bonds issued, the amount of loans attracted annually and the dividends paid out for the ideal values in the first potency matrix are summarized in tables 4 to 6 below. Here, it can be seen that the bond issues are relatively unattractive for the realization of high dividends for financing with one-period loans the reverse argument holds.

	Goal Variable					
Bonds	Pv-Div	Dismiss	SensOil-	SensOil+	SensWage	SensInt
0		0.03	1	0.53	1	1
1		1	1	1	0.86	1
2		1	0.86	1	1	
3		1	1	1	0.55	1

Table 5: Bond issues when different goal variables are optimized.

	Goal Variable					
Dividends	Pv-Div	Dismiss	SensOil-	SensOil+	SensWage	SensInt
0	452	619	667	664	681	516
1	498	557	601	507	613	540
2	548	501	541	538	551	480
3	493	451	486	483	496	440
4	542	455	438	436	446	483
5	596	409	394	392	402	435
6	537	3680	355	353	361	411
7	483	331	320	317	325	452
8	434	298	288	286	293	407
9	478	269	259	257	263	448

Table 6: Dividend payments when different goal variables are optimized. (values in $ 1,000)

	Goal Variable					
Loans	Pv-Div	Dismiss	SensOil-	SensOil+	SensWage	SensInt
0	683			60		
1	752	121	93	21	23	
2	615	18	72	12		
3	3068	39	2520	137	29	
4	2663		2183	67		
5	2278	93	1839	125	10	
6	1773	295	1532	268	215	
7	1130	40	893		10	
8	525	245	782	263	215	

Table 7: One-period loans attracted when different goal variables are optimized. (values in $ 1,000)

Starting from the potency matrix in table 3, we have successively raised the pessimistic values for various goal variables. The first lower bound to be raised was the minimal value of the oil sensitivity. The minimal, or in this case, the maximum negative, oil sensitivity was reduced from $ 143,800 to $ 50,000 for a ten percent change in the oil price. This improvement came at the cost of a lower ideal present value of dividends and a higher maximum sensitivity for oil price changes. The range in which the oil sensitivity for each period in the ideal solution would be, was thereby shifted upward. Then, the interest sensitivity was lowered, which caused a further reduction of the present value of the dividend stream and a deterioration of the ideal minimum oil sensitivity. In the following iterations, the minimal levels for the wage rate sensitivity, the present value of the dividends and the number of dismissals are set. In the final iterations, these values are further improved until a set of solutions is obtained which are only marginally different. The potency matrices in all iterations are presented in table 3 which illustrates the 'tightening' of the range between pessimistic and ideal solutions for the goal variables. In figure 4 and 5, this is graphically shown for the first and the sixth goal variable respectively. The final solution is given in table 8 below.

Period	Loans	Dividends
0	116	393
1	188	432
2	0	443
3	602	399
4	409	439
5	269	483
6	159	531
7	7	584
8	0	642
9	0	622

Projects			
		10	1
0	1	11	
1		12	
2		13	
3		14	1
4		15	
5		16	0.525
6	0.274	17	0.235
7	1	18	1
8		19	0.526
9		20	

Table 8: Final solution

5 Operationalization

With respect to the operationalization of the multi-factor framework, a series of problems arise. In this section we will address some of the problems associated with:

a. the calculation of expected values; and

b. the estimation of sensitivities.

Problems related to the identification and specification of projects, the specification and valuation of options and with the use of IMGP will not be dealt with here, because they are of a more general nature and because they are also dealt with elsewhere. In order to calculate the expected cash flows, both of the firm 'as it is' and of the proposed projects a 'base case' or 'expected' scenario has to be defined. That is, a set of assumptions with respect to the expected values of the firm's most important environmental factors together with an overview of the decisions to be made by the firm in case the firm's environment would be as expected. The next step is to calculate the expected cash flows. This can be based on a more or less complicated model, relating the factor values and the decisions to the expected cash flows. A natural starting point for building such a model is a simple cash flow definition, for the firm as a whole and/or for the different divisions of the firm. To estimate the sensitivities associated with the existing firm and with the projects, a wide variety of approaches is available. The degree of detail of the analyses may vary from an enumeration of the relevant factors on the one hand to complex statistical analyses on the other. Along this scale, the information obtained on the factors considered increases. Simple techniques to analyze factor sensitivities are e.g. chartreading, check-lists, qualitative interviews or even group-discussions. With these techniques, the relevance of factors is established in a qualitative manner. Some quantitative information is used in spread-

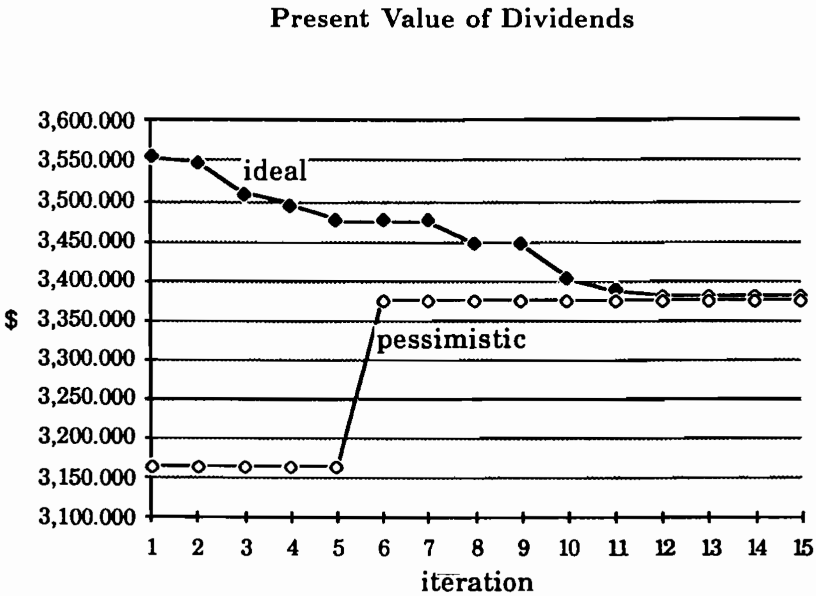


Figure 4: Ideal and pessimistic values for the Present Value of the Dividend stream

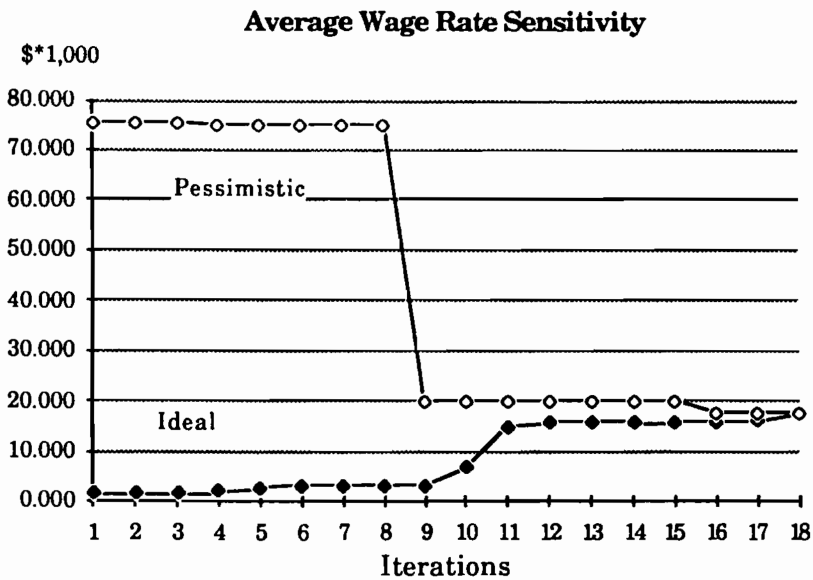


Figure 5: Ideal and pessimistic values for the Interest Sensitivity.

sheet- or simulation-applications, in which not only the presence but also the estimated effect of a factor is incorporated. The effects may be obtained by rules of thumb, by extrapolation of historic data or by more complex estimation procedures. Information of a more quantitative nature may be obtained e.g. by time series analysis. Obviously, this is not without difficulties. For instance, factors influencing the cash flows are seldomly independent of each other. The influence of the factors on the cash flows may be described by means of linear models but is possibly better described through non-linear relations. How to separate expected from non-expected influences? Are the sensitivities to be measured on a monthly basis, on a quarterly basis or something in between? What about the potential instabilities in the sensitivities? And so on. A lot of questions remains to be answered. Nevertheless, the experiences acquired during the work following the paper by Hallerbach and Spronk, 1986, suggest that the multi-factor road is not without difficulties but nevertheless seems to lead to valuable results.

6 Conclusion and remarks

Although thus far, no real application of the proposed approach was made, discussions with practitioners suggest that 'we are on the good way' and have lead to adaptations of the procedures used to evaluate and to select capital investment projects. Clearly, lots of problems are still to be solved, but we strongly believe it to be worthwhile to go through the process of solving them. In this paper, no explicit attention was paid to the case in which financial planning involves two or more hierarchically distinct decision levels. An approach to such two-level financial planning including the multi-factorial approach is discussed in another paper (see Goedhart, Schaffers and Spronk, 1988).

References

Casey, C. J. and Bartczak, N. J. (1984). Cash Flow – it's not the Bottom Line. Harvard Business Review, August-July.

Diepenhorst, A. I. (1962). Beschouwingen over de optimale financiele stuctuur van de onderneming. Inaugurale rede, Bohn, Haarlem, (in Dutch).

Donaldson, G. (1969). Strategy for Financial Mobility. Harvard University Press, Boston.

Donaldson, G. (1984). Managing Corporate Wealth. Harvard University Press, Boston.

Goedhart, M., Schaffers, J. and Spronk, J. (1988). An Interactive Multi-Factor Procedure for Two-Level Financial Planning with Conflicting Goals. Report 8708/f, Centre of Research in Business Economics, Erasmus University, Rotterdam.

Hallerbach, W., Spronk, J. (1986). An Interactive Multi-Factor Portfolio Model. Report 8610/f, Centre for Research in Business Economics, Erasmus University, Rotterdam.

Kemna, A. G. Z. (1988). Options in Real and Financial Markets. PhD thesis, Erasmus University Rotterdam, Donner Boeken, Rotterdam.

Schaffers, J. Spronk, J. (1987). Two-Level Financial Planning with Conflicting Goals, An Interactive Procedural Approach. In: Sawaragi, Y., Ihoue, K., Nakayama, H.(eds), Towards Interactive and Intelligent Decision Support Systems, Volume 1, Springer, Heidelberg, pp. 270–279.

Spronk, J. (1981). Interactive Multiple Goal Programming : Applications to Financial Planning. Martinus Nijhoff, Boston.

Spronk, J. (1985). Financial Planning with Conflicting Objectives. In: Fandel, G. and Spronk, J.(eds), Multiple Criteria Decision Methods and Applications, Springer, Berlin, pp. 269–288.

Spronk, J. (1989). Multi-Factorial Financial Planning, In: Lockett, A. G. and Islei, G.(eds.), Improving Decision Making in Organisations, Springer, Heidelberg.

Appendix 1

	SensOil									
Project	0	1	2	3	4	5	6	7	8	9
0	-50.0	-70.0	-60.0	-30.0	-40.0	-50.0	-60.0	-30.0	-40.0	-50.0
1	-6.0	-6.0	-6.0	-6.0	-8.0	-8.0	-8.0	-8.0	-8.0	-8.0
2	-8.0	-8.0	-8.0	-9.0	-8.0	-10.0	-10.0	-10.0	-10.0	-10.0
3	-19.0	-19.0	-19.0	-19.0	-20.0	-20.0	-20.0	-19.0	-19.0	-20.0
4	-17.0	-17.0	-17.0	-17.0	-17.0	-17.0	-17.0	-17.0	-17.0	-17.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	15.0	15.0	15.0	15.0	15.0	15.0	15.0
7	0.0	5.0	5.0	5.0	5.0	0.0	0.0	0.0	0.0	0.0
8	-8.0	-8.0	-8.0	-8.0	-8.0	-8.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.0	6.0	6.0
10	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0
11	-10.0	-10.0	5.0	5.0	5.0	0.0	0.0	0.0	0.0	0.0
12	0.0	17.0	17.0	17.0	17.0	17.0	0.0	0.0	0.0	0.0
13	0.0	0.0	-3.0	-3.0	-3.0	-3.0	0.0	0.0	0.0	0 .0
14	0.0	12.0	12.0	12.0	12.0	12.0	12.0	12.0	12.0	12.0
15	20.0	20.0	20.0	20.0	20.0	-5.0	-5.0	-5.0	-5.0	-5.0
16	-6.0	-6.0	-6.0	-6.0	0.0	0.0	0.0	0.0	0.0	0.0
17	25.0	25.0	25.0	25.0	25.0	25.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	-9.0	-9.0	-9.0	-9.0	-9.0	-9.0	0.0
19	10.0	10.0	10.0	10.0	15.0	15.0	15.0	15.0	15.0	15.0
20	-10.0	-10.0	-10.0	-10.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0

Table A1: Sensitives of Annual Project Net Cash Flow for Unexpected Changes in Oil Price Level
(changes in $ 1,000 for 10% change in oil price)

	SensWage									
Project	0	1	2	3	4	5	6	7	8	9
0	-50.0	-40.0	-40.0	-40.0	-40.0	-30.0	-30.0	-20.0	-20.0	-20.0
1	-25.0	-25.0	-25.0	-25.0	-20.0	-25.0	-15.0	-25.0	-10.0	-5.0
2	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0	-1.0	-1.0	-1.0	0.0
3	-5.0	-5.0	-5.0	-10.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0
4	-6.0	-6.0	-6.0	-6.0	-6.0	-6.0	-6.0	-6.0	-6.0	-6.0
5	-10.0	-10.0	-10.0	-10.0	-10.0	-10.0	-10.0	-10.0	-10.0	-10.0
6	0.0	0.0	0.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0
7	0.0	-5.0	-5.0	-5.0	-5.0	0.0	0.0	0.0	0.0	0.0
8	-7.0	-7.0	-7.0	-7.0	-7.0	-7.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	-9.0	-9.0	-9.0
10	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0	-15.0
11	-10.0	-10.0	-5.0	-5.0	-5.0	0.0	0.0	0.0	0.0	0.0
12	0.0	-18.0	-18.0	-18.0	-18.0	-18.0	0.0	0.0	0.0	0.0
13	0.0	0.0	-9.0	-9.0	-11.0	-11.0	-11.0	0.0	0.0	0.0
14	0.0	-12.0	-12.0	-12.0	-12.0	-12.0	-12.0	-12.0	-12.0	-12.0
15	-18.0	-18.0	-18.0	-18.0	-18.0	-18.0	-2.0	-2.0	-2.0	-2.0
16	-13.0	-13.0	-10.0	-10.0	0.0	0.0	0.0	0.0	0.0	0.0
17	-20.0	-20.0	-20.0	-20.0	-20.0	-20.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	-19.0	-19.0	-19.0	-19.0	-19.0	-19.0	0.0
19	-9.0	-9.0	-9.0	-9.0	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0
20	-9.0	-9.0	-9.0	-9.0	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0

Table A2: Sensitivities of Annual Project Net Cash Flow for Unexpected Changes in Wage Level
(changes in $ 1,000 for 10% change in wage rate)

A Study of the Effect of Interactive Preference Aggregation in a Multiobjective Group Decision Aid

Peri Iz
Department of Decision Sciences
Miami University
Oxford, Ohio 45056, USA

M. Tawfik Jelassi
Technology Management Area, INSEAD
77305 Fontainebleau, France

Abstract

This paper reports on the results of a laboratory experiment involving the use of a decision support tool in finding a group compromise solution to a multiobjective problem. The effect of two independent variables on subjective and objective performance measures are investigated. The presence or absence of a formal preference aggregation procedure in the decision support tool and the level of the decision maker's linear programming background are the two independent variables. Solution quality, speed of convergence to a final agreement, and user confidence in the best compromise solution constitute the dependent variables. Analysis and implications of this study results are provided and future research work outlined.

1 Introduction

Despite the increasing popularity of computer-supported MCDM tools (Jelassi et al., 1985; Jelassi, 1987), the effectiveness of such procedures when used by multiple decision makers (DM) remains unproven. In group decision making the preferences of the group members are expected to vary from each other. Consequently, determining the best alternative from a given set of solutions requires aggregation of individual preferences. This is especially true for an interactive procedure which would require group feedback to generate alternative solutions.

The preference ordering procedure suggested by Cook and Kress (1985) and the consensus ranking method developed by Beck and Lin (1983) are used to aggregate individual preferences in this group decision support aid. Although the integration of MCDM techniques with a preference aggregation procedure have been suggested by other authors

(Henderson and Schilling, 1985; Lewandowski et al., 1986; Lewis and Reeves, 1987), the improvement in the effectiveness of group decision making due to the addition of a preference aggregation method have not been empirically tested before.

Section 2 of this paper describes the empirical study, namely the group decision problem, experiment hypotheses, and research methodology. Section 3 analyzes the results and discusses the implications of the study findings. Section 4 concludes this paper with some suggestions for future research work.

2 Empirical study

The task used in this study involves finding a group compromise solution to an aggregate production planning problem with three conflicting objectives. Although the aggregate planning problem itself is relatively well-structured, the existence of multiple DMs with different priorities concerning the three conflicting objectives makes the use of a group decision support procedure very attractive.

Aggregate production planning is an important task for an organization. It is used to determine the production, inventory, and work force levels at a product group or family level supportive of the higher level strategic plan. Once this aggregate plan is agreed upon, it is necessary to prepare a detailed master production schedule giving the timing and sizing of the manufacturing of individual products. The next level in the planning hierarchy includes plans for materials and manufacturing activities which are driven by the master schedule.

In this laboratory experiment, four group decision support configurations are studied. Each is created by manipulating two independent variables across two levels. The independent variables are the presence or absence of a formal preference aggregation method in solving the group decision problem and the strength of the DM's linear programming (LP) background. The effect of each configuration is assessed experimentally on three dependent variables: quality of the final outcome, speed of reaching a group compromise, and DM's confidence in the group solution.

The experiment consists of pre-study activities and group sessions. The first phase includes classroom lectures on LP and MCDM concepts, and pretesting to determine the strength of individual backgrounds in this area. In phase two, groups of DMs search for a compromise solution to the aggregate planning problem using a group decision making procedure.

2.1 A model for the group decision problem

Typical objectives of an aggregate production plan are good customer service, minimum inventory investment, and maximum plant operating efficiency. The essence of good customer service is to be able to deliver the product to the customer in the shortest possible time period. This may require available on-hand inventory which contradicts the objective of maintaining minimum inventory investment. On the other hand, one of the most significant aspects of plant efficiency is to keep the plant running at a steady pace to avoid having to hire, train, and lay off people too frequently. Under fluctuating demand this may increase inventory levels at times. Hence, the major objectives of an aggregate production plan are in conflict. Anyone of the objectives can be met by ignoring the others but a successful company would try to meet all three objectives simultaneously and reasonably well.

In this study the theory of multiobjective LP is used as a modeling tool in developing a group decision making procedure. A goal programming model is used to generate solutions to the aggregate production problem. Three conflicting objectives are considered with respect to three functional areas in a company: customer service, stable work force, and profitability. Customer service is the major marketing objective and here it is measured by the number of back orders. The service objective is to minimize the total units of two products back ordered during the year. The second objective minimizes the total changes in the work force from different time periods. The third objective maximizes the difference between the sales revenues and the cost of labor, material, inventory, and overtime production.

The traditional approach of assigning arbitrary values to represent the cost to the company of back orders and work force changes and including them in the profit function is not used. Instead, the service and work force objectives are treated separately. These three conflicting objectives are subject to a set of constraints. The maximum and minimum levels of sales forecasts are specified by the sales limitations. The production constraints limit the level of overtime production and lay offs in different time periods. Finally, two other sets of constraints define the available labor and machine time for each month.

2.2 Hypotheses of the study

The current study compares a set of performance measures from a laboratory testing of a group decision-making procedure that includes a consensus ranking method with similar measures from an informal group decision process. Specifically, the following null hypotheses are explored:

Hypothesis 1: The quality of group compromise solutions found by groups using a procedure that includes a preference aggregation method is higher than the quality of solutions found by groups using the informal approach.

Hypothesis 2: The quality of group compromise solutions found by groups made up of DMs with a strong LP background is higher than the quality of solutions found by groups consisting of DMs with a weak LP background.

Hypothesis 3: Groups using the system that includes a formal preference aggregation method take less time to reach a group compromise solution than those using the informal approach.

Hypothesis 4: Groups consisting of DMs with a strong LP background reach a group compromise solution in less time than those consisting of DMs with a weak LP background.

Hypothesis 5: Groups using the informal decision support procedure will have to make more iterations to generate alternative solutions than their counterparts.

Hypothesis 6: Groups consisting of DMs with a strong LP background will make fewer iterations to generate alternative solutions than those with a weak LP background.

Hypothesis 7: DMs using a group decision support procedure that includes a formal preference aggregation method will have a confidence level in the group compromise solution no different from that of DM using the informal approach.

2.3 Study methodology

A laboratory test based on a simulated business environment was used to evaluate the impact of a computer-supported group decision-making procedure. So far, most of the empirical MIS/DSS research involved individual decisions (see, for example, Aldag and Power, 1986; Benbasat and Dexter, 1982, 1985; Benbasat and Schroeder, 1977; Cats-Baril and Huber, 1987; Chakravarti et al., 1979; Dickmeyer, 1983; Dickson et al., 1977; Dos Santos and Bariff, 1988; Eckel, 1983; Goslar et al., 1986; King and Rodriguez, 1978; McIntyre, 1982). Only a few studies, such as (Iz, 1987), (Joyner and Tunstall, 1970), and (Sharda et al., 1988), examined the effect of decision support aids on dependent measures. Joyner and Tunstall's study (Joyner and Tunstall, 1970) revealed no significant improvement in the quality of decisions made by groups using a computer program called CONCORD. On the other hand, Sharda et al. (1988), report a positive effect due to the use of a group decision tool on performance variables such as profits and volatility. Iz (1987) compared three group decision procedures with respect to a set of objective and subjective measures. The results of this study favor group decision procedures utilizing structured solution models over those using informal strategies.

The current study employed a 2×2 factorial design. The independent variables were: using a computerized group decision aid that includes a formal preference aggregation procedure versus an informal group decision process, and having a strong versus a weak LP background. The dependent variables consisted of effectiveness and efficiency measures such as quality, speed, and DM's confidence in the final compromise solution.

2.3.1 Independent variables

The first variable had two levels, formal versus informal group decision procedure. Groups using the formal procedure were presented, at the end of each iteration, with information related to a group preference solution in addition to the individually preferred solutions. This information was obtained through the following steps: First a set of solutions was found corresponding to the individual goal levels specified for the three objectives in the aggregate planning model. Next, DMs were asked to nominally rank these solutions. Finally, the most preferred group solution was determined using the minimum regret heuristic of Beck and Lin (1983). If the most preferred solution was unanimously voted acceptable, then the process is discontinued. Otherwise, a new set of goal levels were specified by each DM and the above steps were repeated. Groups who were not using the formal procedure searched for a compromise solution by varying the goal levels of the objectives. These groups had to discuss and agree on a set of goal levels at each iteration before they generated a solution in contrast to those who used the formal procedure. The latter group discussed only the most preferred solution found by the minimum regret algorithm after group members studied and ranked the set of solutions corresponding to goal levels specified for the objectives by each DM. The second variable had two levels, strong versus weak LP background. Since the modeling and solution of the aggregate planning problem involved linear programming, the effect on the dependent measures of the subjects' LP background was controlled by this variable.

2.3.2 Dependent variables

Research in the DSS area (e.g. Dos Santos and Bariff, 1988; Rothernal, 1981; Wallenius, 1975) suggest several dependent variables that can be adopted for studying the impact of a group decision aid. In this study, solution quality was used as the primary variable to measure group decision effectiveness. The quality of a final compromise solution is determined by the following average percentage achievement (a.p.a.) measure:

$$\text{a.p.a.} = \sum_{j=1}^{3} \left[\frac{\left| \frac{Z_j - Z_j^w}{R_j} \right|}{3} \right] \tag{1}$$

where,

Z_j is the value of objective j in a compromise solution.

Z_j^w is the worst value that objective j can achieve. (This can be determined by constructing a payoff matrix for the multiobjective problem); and

R_j is the range of variation in the value of objective j. (This can also be determined from the payoff matrix).

Efficiency of group performance was measured by the time required to reach a compromise solution. In addition to the objective dependent measures listed above, a hundred point Likert-type scale was used to measure the confidence of the DMs in the final compromise solution.

2.3.3 Subjects

The experimental subjects in this study were junior and senior business students enrolled in an introductory level operations research course. Subjects participated in the laboratory experiment to fulfill one of the course requirements. Twenty-five percent of each subject's final grade was based on the score he/she received from the outcome of this experiment.

2.3.4 Decision task

The decision task involved a course project which required a group of three subjects to find a compromise solution to an aggregate planning problem with three conflicting objectives. A pilot study using seven groups of students was used to test the complexity of the task and to modify the steps of the experimental procedure for the main study.

	Strong L.P. Background	Weak L.P. Background
Informal Procedure	Configuration 1	Configuration 2
Formal Procedure	Configuration 3	Configuration 4

Table 1: Configurations in the experimental design

2.3.5 Experimental procedure

The sequence of events that *took* place in the current study were as follows:

Pre-study events: These events consisted of classroom lectures on multiobjective decision making, assignment of the aggregate production planning model as a class project to be formulated by the subjects, administration of an in-class test to determine participants' LP understanding, assignment of subjects to groups, and finally distribution of a written description of each subject's role in the experiment.

The course material prior to the experiment covered formulation and solution of LP models which was followed by a closed-book, in-class test. Test results were used to determine those subjects with strong and weak LP backgrounds. The tests were graded on a scale of 0 to 100. Subjects who scored higher than 90 points were considered as having a strong LP background; those scoring lower than 70 points were categorized as having a weak LP background. Data collected from those subjects with grades between 70 and 90 were not used in the analysis.

The assignment of subjects to groups was based on their LP level. Three subjects with similar LP backgrounds were assigned to the same group. Although not always possible, every effort was made in this selection process to include within a given group a subject majoring in a functional area such as finance, marketing, and management.

Following the test, subjects had two weeks of classes introducing the basic concepts underlying multiobjective programming methods. A brief description of the goal programming method and its application to small example problems were presented during these classes. The first phase of the project required each subject to formulate the multiobjective problem used in this study. This project was assigned after two weeks of classes on MCDM and had to be turned in for grading a week later. The final pre-study event was a meeting with each group during which subjects were asked to decide the managerial role they would like to play during the experiment.

Group Sessions: This study investigates the effect of fourconfigurations created by manipulating two independent variables group decision methods and LP background. Table 1 shows these four experimental treatments.

Configuration 1 involves groups of students with a strong LP background. Each group had to find a compromise solution to the aggregate planning problem using the informal group decision procedure. Configuration 3 also consisted of groups of subjects with a strong LP background. However, they searched for a compromise solution using the

Effect	Dependent Variables	F
Type of Group Decision Procedure	Quality	7.42**
	Time	0.25
	Iterations	40.69*
	Confidence	24.05*
Linear Programming Background	Quality	0.26
	Time	3.60**
	Iterations	7.47***
	Confidence	0.47

$^{*}p < 0.01$
$^{**}p < 0.05$
$^{***}p < 0.10$

Table 2: ANOVA Results

group decision support aid which included a formal preference aggregation procedure. Configurations 2 and 4 are counterparts of Configurations 1 and 3 respectively where groups consisted of subjects with a weak LP background. These group sessions were held in a computer laboratory equipped with terminals and a printer. Each group session was limited to ninety minutes after which each subject was asked to rate his/her confidence in the final solution on a 100-point Likert scale.

3 Analysis of results and implications of the study

A two factorial fixed effects ANOVA model was used to determine the effect of the independent variables on the dependent measures. A total of twenty-three groups participated in the experiment. The form of the model is as follows:

$$\begin{array}{l}\text{Dependent}\\ \text{Measures}\\ \left[\begin{array}{l}\text{Quality}\\ \text{Time}\\ \text{Iterations}\\ \text{Confidence}\end{array}\right]\end{array} = \mu\ldots + \alpha\left[\begin{array}{c}\text{Type of}\\ \text{Group Decision}\\ \text{Procedure}\end{array}\right] + \beta\left[\begin{array}{c}\text{Linear}\\ \text{Programming}\\ \text{Background}\end{array}\right] \tag{2}$$

The usual assumptions of the fixed effects model were made and indirect tests for model adequacy were performed by plotting and examining the behavior of residuals against each variable. The results from these plots were in favor of the above model. Table 2 summarizes the ANOVA results.

Independent Variables		Dependent Variables			
	Levels	Quality	Time	Iterations	Confidence
Group Decision Procedure	Formal	0.803**	0.854	2.625*	93.750**
	Informal	0.724	0.906	8.750	80.625
LP Background	High	0.771	0.980**	7.000	88.125
	Low	0.756	0.780	4.375***	86.250

$^{*}p < 0.01$
$^{**}p < 0.05$
$^{***}p < 0.10$

Table 3: Cell means for the main effects

3.1 Solution quality

The first two hypotheses dealt with the performance related effects of the independent variables. Hypothesis 1 stated that the quality of compromise solutions found by the formal group decision procedure would be higher than that found by groups using the informal approach. The results (shown in Table 3) indicate that the average percentage achievement of the ideal solution for those groups using the formal approach was higher than their counterparts using the informal approach.

Hypothesis 2 stated that the quality of group compromise solutions found by DMs with a strong LP background would be higher than that found by DMs with a weak LP background. As indicated in Table 3, this hypothesis was not supported by the data. The degree of knowledge about the solution method employed by either approach did not play an important factor in reaching a compromise solution closer to the ideal solution.

3.2 Time required to reach a group compromise solution

Hypothesis 3 claimed that groups using the formal approach would take less time to reach a compromise solution. The results in Tables 2 and 3 show no significant effect on time due to the type of group decision procedure used. These results may be partly due to the nature of this study. Although it was explained to the subjects that the length of time they spend on the project will not necessarily affect their grades, students may have still expected time to be an important determinant of their project grade.

Hypothesis 4 posited that groups consisting of DMs with a strong LP background would reach a compromise solution faster than their counterparts. As indicated in Tables 2 and 3, the level of LP background had a significant but directionally opposite effect than that which was expected on the time measure. This finding may be explained by the fact that subjects in the high LP groups were in general better students and therefore put more time and effort into their projects.

3.3 Number of iterations

Hypothesis 5 was supported by the data. Results in Table 2 show that the type of group decision procedure did have a significant effect on the number of iterations. As hypothesized, the formal group decision procedure required groups to make fewer iterations in generating alternatives. These groups were able to study and rank solutions related to other DMs' priorities. Their discussions were centered around the solution which was the least regretted by group members. Therefore, better compromises were expected by these participants than their counterparts using the informal approach.

Linear programming background had a significant effect but opposite in the direction stated by Hypothesis 6. Groups consisting of DMs with a strong LP background needed more iterations to find a compromise solution. This finding implies that the efficiency of a model-based group decision support procedure, such as the one employed in this study, is independent of the group members' depth of knowledge on the particular method used for generating alternative solutions.

3.4 Confidence

Hypothesis 7 addressed the group members' confidence in the final compromise solution based on the approach being used. Contrary to what was hypothesized, confidence of DMs using the formal procedure was significantly higher than that of DMs using the informal approach (Table 3). This result seems to be conflicting with the recent findings of Sharda, et al. (1988). However, the preference aggregation component of the formal approach used in this study requires the groups to reevaluate the most preferred alternative in every iteration. This can lead DMs at the end to believe that significant effort has been made to find a group compromise solution in which they can place a high level of confidence.

4 Conclusions and future research

The results of this study indicate that groups using a decision support aid with a formal preference aggregation approach reached higher quality solutions in approximately the same amount of time but with fewer iterations than their counterparts using the informal procedure. In fact, of the four dependent measures, time required to reach a compromise solution was the only measure with respect to which the two group decision procedures were not significantly different. These results are in agreement with earlier studies by Iz (1987) and Sharda, et al. (1988).

So far, the limited amount of empirical evidence in this area indicates that the level of structure in a group decision procedure is a contributing factor in higher decision making performance, but further research is necessary to generalize these results. Similar experiments should be performed for solving different group decision problems using other MCDM models. Systematic variation of the decision task, solution method, and preference aggregation strategy can be more informative in determining appropriate decision support for different group settings. However, it is very likely that preference for using a particular group decision procedure will be in part determined by the decision making environment at hand. For example, a context that emphasizes research and long-term strategic planning may require different group decision procedures from an environment in which frequent and quick analyses are needed.

Although the subjects in the present study only assumed the role of executives in an actual company, each participant had an interest in the performance of their group. Twenty-five percent of the subjects' overall grade on the course was determined by their performance in the laboratory experiment. Prior to the experiment, each subject was required to formulate an LP model for the aggregate planning problem. This step not only determined part of the subjects' project grade, but it also familiarized them with the constraints and the conflicting objectives of the group decision problem. The final step before the laboratory experiment provided the participants with detailed information about the particular functional area for which they are responsible through individual scenarios.

Finally, the effect of other analytical decision tools, such as simulation, must be investigated in a group decision environment besides multicriteria optimization on which this study has focused.

References

Aldag, R.J. and Power, D.J. (1986). An Empirical Assessment of Computer-Assisted Decision Analysis. *Decision Sciences*, Vol. 17, No. 4.

Beck, M.P. and Lin, B.W. (1983). Some Heuristics for the Consensus Ranking Problem. *Computers and Operations Research*, Vol. 10, No. 1.

Benbasat, I. and Dexter, A.S. (1982). Individual Differences in the Use of Decision Support Aids. *Journal of Accounting Research*, Vol. 20, No. 1.

Benbasat, I. and Dexter, A.A. (1985). An Experimental Evaluation of Graphical and Color-Enhanced Presentation. *Management Science*, Vol. 31, No. 11.

Benbasat, I. and Schroeder, R.G. (1977). An Experimental Investigation of Some MIS Design Variables. *MIS Quarterly*, Vol. 1, No. 1.

Cats-Baril, W.L. and Huber, G.P. (1987). Decision Support Systems for Ill-Structured Problems: An Empirical Study. *Decision Sciences*, Vol. 18, No. 3.

Chakravarti, D., Mitchell, A.A. and Staelin, R. (1979). Judgment-Based Marketing Decision Models: An Experimental Investigation of the Decision Calculus Approach. *Management Science*, Vol. 25, No. 3.

Cook, W.D. and Kress, M. (1985). Ordinal Ranking with Intensity of Preference. *Management Science*, Vol. 31, No. 1.

Dickmeyer, N. (1983). Measuring the Effects of a University Planning Decision Aid. *Management Science*, Vol. 21, No. 6.

Dickson, G.W., Senn, J.A. and Chervany, N.L. (1977). Research in Management Information Systems: The Minnesota Experiments. *Management Science*, Vol. 23, No. 9.

Dos Santos, B.L. and Bariff, M.L. (1988). A Study of User Interface Aids for Model Oriented Decision Support Systems. *Management Science*, Vol. 34, No. 4.

Eckel, N.L. (1983). The Impact of Probabilistic Information on Decision Behavior and Performance in an Experimental Game. *Decision Sciences*, Vol. 14, No. 4.

Goslar, M.D., Green, G.I. and Hughes, T.H. (1986). Decision Support Systems: An Empirical Assessment for Decision Making. *Decision Sciences*, Vol. 17, No. 1.

Henderson, J.C. and Schilling, D.A. (1985). Design and Implementation of Decision Support Systems in the Public Sector. *MIS Quarterly*, Vol. 9, No. 2.

Iz, P. (1987). A Comparative Study of Three Group Decision Procedures for Multiobjective Problems. Ph.D. Dissertation, The Ohio State University.

Jelassi, M.T., Jarke, M. and Stohr, E.A. (1985). Designing A Generalized Multiple Criteria Decision Support System. *Journal of Management Information Systems*, Vol. 1, No. 4.

Jelassi, M.T. (1987). MCDM: From 'Stand-Alone' Methods to Integrated and Intelligent Decision Support Systems. In: Inoue, K., Nakayama, H. and Sawaragi, Y. (eds.), Interactive and Intelligent Decision Support Systems, Vol. 2, Berlin: Springer-Verlag.

Joyner, R. and Tunstall, K. (1970). Computer Augmented Organizational Problem Solving. *Management Science*, Vol. 17, No. 4.

King, W.R. and Rodriguez, J.I. (1978). Evaluating Management Information Systems. *MIS Quarterly*, Vol. 2.

Lewandowski, A., Johnson, S. and Wierzbicki, A. (1986). A Prototype Selection Committee Decision Analysis and Support System, SCDAS: Theoretical Background and Computer Implementation. IIASA Working Paper WP-86-27, Laxenburg, Austria, 1986.

Lewis, H.S. and Reeves, G.R. (1987). An Interactive Procedure for the Multiple Decision Maker Multiple Objective Decision Problem. Proceedings of the National Meeting of the Decision Sciences Institute, Boston, Massachussets.

McIntyre, S. (1982). An Experimental Study of the Impact of Judgment-Based Marketing Models. *Management Science*, Vol. 28, No. 1.

Rothernal, M.A. (1981). A Study of Three Methods of Eliciting Preference Information for Use in a Multiple Objective Decision Making Model. Ph.D. Dissertation, Department of Management Sciences, The Ohio State University.

Sharda, R., Barr, S.H. and McDonnell, J.C. (1988). Decision Support System Effectiveness: A Review and an Empirical Test. *Management Science*, Vol. 34, No. 2.

Wallenius, J. (1975). Comparative Evaluation of Some Interactive Approaches to Multicriterion Optimization. *Management Science*, Vol. 21, No. 12.

MIDA: Experience in Theory, Software and Application of DSS in the Chemical Industry

J. Kopytowski *
Industrial Chemistry Research Institute ,
M. Zebrowski
Joint Systems Research Department
of the Institute for Control and Systems Engineering,
Academy of Mining and Metallurgy, Cracow,
and the Industrial Chemistry Research Institute, Warsaw

Abstract

DSS and methodology for programming development of the chemical industry called MIDA or Multiobjective Interactive Decision Aid have been developed and since then are extensively applied.

It was done already in a variety of cases and in diverse decision as well as cultural environments. On the verge of the second decade of this type of activity, an experience in theory, software and application is presented. It is aimed at pointing out important aspects in all spheres of the activities considered. The paper covers perspective and scope of the development programming domain showing how the identification of the field opened way to establishing of a theoretical and methodological framework and to consequent development of the MIDA system, its architecture and software development. The experience goes beyond the particular field of MIDA application and seems to be generally meaningful and therefore useful in development and application of various decision support tools and systems [1].

1 Introduction

1.1 Origin and motivation

On the verge of the second decade of the research work in the area of Decision Support Systems, concentrated on a methodology for Programming Development of the Chemical Industry, an experience gained so far calls for synthetic review and presentation.

The unremitting search for an efficient development strategy constitutes a vast, fundamental and vital task of management of practically every industry in today's fast changing

*at present with UNIDO, Vienna.

[1] also published in Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki eds., Lecture Notes in Economics and Mathematical Systems, Vol. 331, Springer-Verlag 1989.

world. This was the most strongly formulated motivation behind the research reported initially in our paper (Gorecki, Kopytowski and Zebrowski, 1978). The motivation not only persisted since then but perhaps became even stronger through all that time. In fact it is very effectively stimulated by a tremendous progress in the DSS development and applications decisively supported by even more rapid and overwhelming developments in computer hardware and software.

This paper is a modified version of the paper *"MIDA: Experience in Theory, Software and Application of DSS in the Chemical Industry"*. It was prepared for the International Conference on Multiobjective Problems of Mathematical Programming, Yalta 26 October - 2 November 1988.

The role of this paper is twofold. First, it is an introduction to the series of six papers which summarizes results of research and application in the domain of MIDA DSS for programming development of the chemical industry. Second, which is perhaps more important, it is to provide a synthetic presentation of key issues and experience gathered so far which go beyond a particular DSS and its applications. Hence the experience seems to be generally applicable in the Decision Support domain.

The whole series of six papers displays in detail selected problems outlined, in this paper, which is an introductory one.

1.2 The scope of the paper

The paper is organized as follows. First, perspective and scope of the programming development of the chemical industry is presented. It is aimed at presentation of key identification issues which determined development of MIDA system and the methodology. It is followed by a briefly commented list of major and recent applications of MIDA to present decision problems and the scope of services performed. Based on this, a synthetic review of the experience in theory, software and applications is given. Conclusions and prospects for the future research are also provided.

1.3 Organization and implementation of the research

The organization and implementation of the research has been developed from its beginning in 1977 and is still in process since the research is an open-ended one in this problem- and application-oriented task. This is reflected in the origin of Joint System Research Department (JSRD) which is a joint venture of the academic institution, namely Institute for Control and Systems Engineering in the Academy of Mining and Metallurgy (ICSE - AMM) and strictly application-rooted organization, i.e. Industrial Chemistry Research Institute (ICRI). The problem-oriented approach can be opposed to the rather common practice which is based on a setting of a general goal: development of DSS for Multiobjective Decision Problems. The latter situation which may be called a tool-oriented approach as opposed to the problem-oriented one can prove its effectiveness in general research aimed at purely scientific results. It is perhaps less effective when tangible effects of a real application are to be reached.

It should be greatly appreciated that in all phases of the work, especially in the course of the identification and the applications, highly qualified experts have been taking part.

These experts represent an experience of high level decision makers combined with a deep technological and economic knowledge. Only this kind of a background combined with the skills of much younger scientists specializing in systems analysis, optimization and control theory as well as computer science could produce a special synergism effect boosting the programs of the work.

1.4 Collaboration with IIASA

Here an important role of collaboration with IIASA must be called into light. It provided a very important platform in MIDA development. The collaboration, due to the type of tasks undertaken by JSRD has been very intensive and involved during last eight years a pretty number of projects and programs which are to be named here. It is to be done not only for the sake of courtesy but to show how wide is the range of problems covered when a DSS such MIDA is to be developed along the way it was assumed to be done.

The most important was the collaboration with SDS or System and Decision Science Program. Especially under chairmanship of A.P. Wierzbicki it provided mutual exchange of approaches and experience. As particularly useful for development of MIDA we evaluate the reference point concept (Wierzbicki 1980) and collaboration at early stages of DIDAS (Lewandowski 1982) development and especially appreciate a close collaboration with A.Lewandowski (Dobrowolski et. al. 1982).

An intensive work was done in parallel for REN or Resources and Environment Area (under J.Kindler) (Dobrowolski et al. 1984). An extensive exchange of scientific contacts was continued through all that time with the Energy Program. Important effects were also gained from the work for Sustainable Development of the Biosphere Program under Bill Clark (Zebrowski and Rejewski, 1987)

These three collaborative links helped to broaden and deepen the identification effort with good effects, especially for the methodological progress. Last but not least an important gain in the software development experience is owing to collaboration with ACA or Advanced Computer Applications program. This also led to valuable applications (Zebrowski et al. 1988).

Concluding, one should emphasize that the variety of approaches and problems tackled in this collaboration played an important role in the progress achieved in seemingly narrow field of development and application of MIDA, and contributed to the results which can be generalized.

2 Scope and perspective of development programming

2.1 Chemical industry and management of change

The world is permanently passing through a chain of great economic, social and technological changes. Recognition of this fact and of the need to control the forces of change has stimulated world-wide interest in the problems of change and methods for coping with them.

Nowhere is the need for management of change more crucial than in the industrial sector, where many factors can affect the growth or decline of individual industries and the resulting industrial structure. The process of change with perhaps the highest impact affects the chemical industry.

Here we concentrate on management of the chemical industry due to the problems it faces as a result of global change, particularly as a result of changing patterns of raw materials and energy use.

The importance of the chemical industry is often greatly underestimated. Not only does it provide soaps, detergents, medicines, but also pesticides, fertilizers, synthetic rubbers, plastics, synthetic fibers... - in fact, our modern technological society could be said to be founded on the chemical industry.

One of the most surprising facts about this industry is that a large proportion of its many products are derived from only a very small number of starting materials, of which hydrocarbons are probably the most important.

As the processing of natural resources with mineral or agricultural origin proceeds, the chains are branching with each processing phase from one generation of intermediates to another. So the developed chemical industry presents in fact ever growing network of interlinked technologies. Final or market goods originating from this network provide only a small share of the total chemical production which does not exceed 25% of the final turnover of the industry.

A practical goal has been to develop a methodology capable of proposing possible restructuring and/or structuring of various sectors of the chemical industry (see e.g. Borek et al., 1978, Gorecki et al.,1978). The approach chosen takes into account a variety of interrelated and alternative production processes (either in use or under development), compares their efficiency, their consumption of different resources etc. and finds the combination of technologies that best meets particular needs while staying within the limits imposed by the availability of resources and environmental constraints.

2.2 Programming development - MIDA approach

From the above essential overview two spheres to be identified emerge. First is the sphere of the present and forecasted performance of the industry which is a result of the identification. It should be described formally in order to represent the changes that transform the industrial structure in time. Second is the sphere of management of the changes where decisions are to be worked out and a decision support is to be developed and naturally embedded into the decision process.

It means that a basic model of industrial structure must be created that reflects the first sphere. It must be based on important assumptions regarding the decision and its corresponding aggregation level as well as boundaries of the industrial structure considered. If the management of change is going to be executed through programming development, then such an activity can be considered as a process of design of an Industrial Development Strategy (IDS).

IDS design is considered as a decision process based on generation of efficient development alternatives expressed in terms of goals, critical or indispensable resources as well as selected array of technologies which are to be utilized. The alternatives are to be generated, selected and ranked along assumed efficiency measures.

The aim of the development programming or IDS design is a selection of the alternative which is to change the industrial structure by means of investment over the time. Due to the dynamic properties of the development process, and specifically the development cycle of technology (Dobrowolski et al. 1985), the time span under consideration is of the range of 10 - 15 years. The straightforward conclusion is that due to the dynamic nature of the development process, IDS design is to be treated and solved as a dynamic problem. It has to be strongly underlined, however, that any attempt to formulate a general multidimensional dynamic problem as a means for generating feasible development alternatives must lead to oversimplification and severe loss of important factors which should not be overlooked. At the same time, any decomposition must assure that through a coherent methodology all the subproblems can be solved as integral parts of the same system. A fundamental premise for the phenomena of development is the fact that as time perspectives become longer (5,10,15... years), the reliability and accuracy of data describing the future decreases.

To meet the challenge of a real application in a complex decision environment, a method better responding to a managerial practice was elaborated. It is based on a decomposition of this in fact dynamic problem along space and time. This will be briefly presented further on after describing substantial results obtained from identification. So before going into discussion of the kind of properties of a decision problem (or problems) that are to be formulated and solved, let us make a step further in the identification of IDS design.

To perform IDS design with focus on generation of development alternatives and their selection, following elements are to be considered:

- Existing industrial structure in terms of consumption coefficients capacities and relevant economic data,
- Potentially available technologies for construction of new plants,

The above two categories form a technological repertoire out of which a new industrial structure is to be devised providing that a harmony between existing elements and the new ones must be sustained. The next category of elements for analysis are:

- resources which are to be utilized in order to implement a new structure, such as investment, manpower, water, energy etc as well as resources which are to be supplied as feedstock to run the new structure.
- some of the resources considered are selected in a special way and called critical resources due to the fact that their availability is a necessary condition to make development alternative feasible.

Critical are those resources which are nominated as such by the decision maker for either being particularly scarce or difficult to obtain; examples may be crude oil, manpower, energy or capital. In practice the set of critical resources is closely related to the set of criteria, since the aim is to find an optimal solution with respect to all critical resources. Technological constraints are quite easily identified and are related to factors such as production capacities and operating conditions. All other elements in the analysis, such as demand for a particular product, the availability of (noncritical) raw materials, fall into the category of complementary or auxiliary information which describes environment to the industrial activities such as terms of trade - specifically prices. On the contrary, a demand for selected significant products as well as availability of selected significant feedstock falls into the category of critical resources.

It is clear that whether a resource is selected as critical or not, it depends on the formulation of the decision problem. In fact, a resource can be nominated to one category or to the other by the decision maker and that in a simple way assures flexibility of predecision analysis since each reassignment to or from the list of critical resources corresponds to a redefinition of a decision problem..

With the above background we can now define *the task of IDS design or generating efficient development alternatives as a quest for concordance between available resources and technologies.*

The state of concordance is to be evaluated along well defined rules and measures for evaluation (and selection) of the efficiency of achieving goals (outputs) from resources supplied to industrial structure (inputs). Such rules and measures form a model of efficiency evaluation which is to be established in order to solve the quest for concordance to yield a feasible development alternatives. This problem area is dealt with in (Zebrowski, 1988).

Technological repertoire, critical resources, constraints and other factors describing the problem (or a particular industrial situation), are to be mapped into second or PDA model of a technological network.

Since it is intuitively obvious that such a process is to be performed through the generation and analysis of multitude of alternatives and their selection, then a mechanism is to be provided that enables to handle the situation. The appropriate models and means for handling the problem of quest for concordance may be organized into a system which is called simply DSS or Decision Support System.

The above philosophy stands behind development of MIDA system and the methodology. In the next step we shall present the assumptions used and a decomposition of the development problem which were applied for practical implementation of the above philosophy.

The effective approach taken in MIDA methodology in the practical implementation of the idea of the quest for concordance can be presented as follows. With respect to the decision level of the programming development, MIDA locates the IDS design on a level which could be named an intermediate economy level (Dobrowolski et al, 1985, Zebrowski 1987). It goes between a macroeconomic level and microeconomic or corporate level. The first one proves to be too aggregated; a single technology cannot be considered in the analysis. Therefore a selection and assessment of appropriate technologies cannot be done at the macroeconomic level. On the other hand, a corporate or enterprise level also proves to be inadequate. This level is too narrow and particular to comprise complex technological and economic relationships which interact in the development process in the chemical industry.

By identifying, defining and choosing the intermediate level, as operational one for programming development, an important original feature has been decided in MIDA development. It comes together with the choice of an entity for setting the feasible scope of the industrial problems which may be regarded as a basic object of the decision analysis. It was called PDA or Production Distribution Area. It helped to respond to the necessity of a formal modeling of industrial structure (IS) of the chemical industry. In fact, this subject is formally described in (Dobrowolski, Zebrowski, 1988).

Due to a possibility of simple aggregation and desaggregation of the elements described in terms of the PDA model, and the PDA level can be split into several levels along so-called problem hierarchy (Gorecki et al. 1983, Dobrowolski et al. 1985). This assures flexibility of the analysis and well corresponds to the industrial practice allowing at the same time to apply MIDA on different levels of aggregation (with data appropriate to the level considered). It makes the concept and application of the PDA model very flexible and rather universal.

From the process of quest as considered so far on the PDA level, a goal structure can be selected representing the assumed state of IS at the end of the horizon covered by the analysis. However, to complete the task of programming, development of a feasible way of transition from the present or actual IS to the selected, final IS is to be optimally selected. The transition is to take into account the following classes of factors:

- technological and market priorities,
- location possibilities,
- construction potential capabilities,
- availability of investment.

In short, to consider these factors, the investment necessary for the transition must be allocated both in space and time.
Therefore three levels emerge and provide a decomposition assumed in MIDA:

- selection of the final or goal IS,
- space allocation of investment,
- time allocation of investment (or investment scheduling).

Appropriate feedback between these levels provides through the space and time allocation a feasibility analysis of the goal structure originally selected.

The three level hierarchy and specifically, the space allocation and investment scheduling levels are discussed both theoretically and practically (through example) in (Skocz, Zebrowski and Ziembla, 1988).

It must be underlined at this point, that the decomposition of the IDS applied in MIDA approach corresponds to the managerial practice. On the other hand, it can be conceptually opposed to more theoretical approaches based on dynamic programming and generally aims at global solution to be obtained from one model (see for e.g. Kendrick 1978, Dobrowolski and Rys, 1981).

The approach applied in MIDA follows from the common decision practice. First the goal "what" must be selected, then questions "where and how" should be answered.

The decisive factor here is that the spatial allocation demands more detailed information related to sites and this must be confronted with spatially disaggregated values obtained from global solution in terms of critical resources. Site specific constraints must be also obeyed.

There is also one more methodological disadvantage coming from globally formulated and solved problems - difficulty of interpretation - especially of cause - effect type. Too many factors are involved at once to enable that kind of analysis. It makes a real interaction with decision maker rather illusory.

3 Evaluation of experience in DSS

3.1 Major MIDA applications

Some most important and representative applications for the scope assumed in this presentation of MIDA system and methodology were selected. The list of applications to be discussed is as follows:

1. **Polish Government Energy Program** - MIDA was used to elaborate a strategy for integration of energy and chemical sectors. MIDA study contributed to the fact that a new development, namely energochemical processing of coal, was brought to light and attained its place in the long-term policy.

2. JSRD competed successfully in offering its services to **UNIDO** and performed the following projects:

   - Master Plan for Development of the Chemical Industry in Iran,
   - Master Plan for Development of the Chemical Industry in Algeria,
   - Master Plan for Development of the Petrochemical Industry in Algeria.

   Within the framework of the above projects the services covered:

   - delivery and installation of equipment and adjacent software as well as delivery and installation of MIDA Decision Support System,
   - training of the counterpart personnel (using lectures, video tapes, top executive seminars and most of all *learning by doing* methodology),
   - elaboration of the development program in various alternatives,
   - industrial and system analysis consulting.

3. **Shanxi Case Study** - this application was done as a part of ACA project in IIASA for Shanxi Province in People's Republic of China and services performed were similar to those described for UNIDO, but the DSS software was developed as a spatially oriented version of the models incorporated in MIDA. The development program for coal based chemical industry was elaborated for the Shanxi province and technical expertise was also shared with the counterpart.

4. **SADCC Study** - *Study of the manufacture of industrial chemicals in the member states of SADCC* - this application was done under the contract with a consulting firm. The firm was contracted for a UNIDO project for SADCC countries. SADCC stands for Southern African Development Coordination Conference. Its members are 9 countries: Angola, Botswana, Lesotho, Malawi, Mozambique, Swaziland, Tanzania, Zambia, Zimbabwe. The consulting company contracted JSRD to perform application of MIDA system for the above study.

The above applications can be categorized to show range of problems and areas that can be tackled with a DSS and methodology such as MIDA as well as to provide a useful generalization of experience aimed.

First category

A problem area related to the development of industrial sectors such as chemical and energy industries is selected. A research is to be carried out and forecasts provided with various technological and development alternatives. This is kind of predecision analysis which includes both research and application type of activities. The responsibility of JSRD as a contracted party covers all the work and study that may be considered as a kind of long range research programs with step by step results to be produced in form of progress and final reports. Results are used by various governmental agencies as well as other scientific centers.

This kind of application is exemplified by no 1 on the above list.

Here a DSS is used by the team performing the job mainly as a laboratory tool. No clearly defined decision maker is present in the process. In such case a variety of skills and experience, specifically presence of industrial experts in the team is especially decisive for good results to be obtained. In such cases by in parallel promoting a work devoted to the problem and a work done on developing methodology and DSS system proves to be fruitful and effective. Such in fact is organization of work assumed by JSRD.

Second category

A development program is to be elaborated for a foreign partner. Such were the applications that were contracted by JSRD with UNIDO. This covers wide span of services and responsibilities. The period assigned for the job is relatively very short : in the range of 1,5 - 3 month.

The DSS is to be delivered and installed together with computer equipment. Moreover, a user's team must be trained in a variety of skills including not only operation of DSS but first of all methodology of its application. These circumstances impose variety of demands which for the lack of space and the type of paper cannot be discussed in details but must be of deep concern. They can be briefly presented as follows.

The principle of operation of a DSS and methods applied should be as clear as possible and as simple as possible at the same time they must eliminate omitting or loss of any essential factors.

A great attention in DSS architecture, functioning and methodology must be paid to facilitate procedures which may help in validating both : simple source data and resulting development alternatives.

Users' involvement is a key factor, both to assure obtaining valuable and useful alternatives that would be accepted for implementation and to establish self-reliance of the users' team (including a decision maker). This can be done through very extensive educational effort and specifically by working out a "learning by doing" methodology. This must be backed also by very clear and well edited documentation supporting all activities as well as results of the project.

If one would like to compare the two above categories of application it could be formulated as follows.

First category provides more scientific and broad approach but is much less demanding in terms of software development, methodology and reliability of the system. On the other hand the second category provides extremely heavy duty testing of all elements taking part in the project.

This includes also all skills and abilities of people involved. It also provides important insights coming from different cultural and decision environments.

Moreover it provides also very useful cases which are an inspiration for the future developments in all aspects : theoretical, software and methodological.

Third category.

The system and methodology are to be adopted for different environment and are to be embedded in another system. Such is the case of the Shanxi case study a work done for ACA IIASA project contracted with Peoples Republic of China.

Apart from the previous remarks formulated for the case of UNIDO projects which remain valid, some additional observations can be formulated.

A DSS becomes a module of a larger system. All kinds of problems of interfacing with other types of software arise. The same concerns interfaces with other models.

At the same time in this particular case new elements specific for spatial allocation backed by scheduling of investment were also developed. In general this kind of applications help finding another way to generalization and standardization of architecture and functionality of the DSS not to mention new theoretical and methodological developments which usually also come in dealing with new, original problems.

Fourth category

The last but not least category is the one when there is no direct contact with the field. The interface comes through third party. It provides a very useful kind of verification of system and methodology. It was the case of the fourth application listed above.

The experience gained so far from a single case reported here may be too limited to be generalized but due to difference in approaches and experience represented by the third party which is supposed to be professional in the field of programming development, a new light can be brought on the own approach which has to defend itself in such circumstances. In fact it also helps to test and improve system and methodology with procedures for validation of data and results.

The above remarks summarize briefly experience in the domain of DSS as gained from major, categorized for that purpose applications of MIDA. Generality of categorization as well as of the relevant experience prove to be useful not only for a specific DSS such as MIDA.

Following this line it seems to be worthwhile to further synthesize the experience and knowledge gained both from application and research point of view.

We may continue with general methodological and theoretical aspects of Decision Support after presenting some selected theoretical and software developments of MIDA system and methodology. Then, this will lead to conclusions and prospects for the future.

3.2 Theory and software

The substantial and perhaps decisive effort was devoted to the identification of the chemical industry, and the mechanisms behind its development (Kopytowski et al. 1982). Special emphasis was given to emerging from this concept of an industrial structure and its properties.

This led to development of a basic model and methodology (Dobrowolski et al. 1984). By proposing the concept of locating the programming development activity, as one executing the management of change on the intermediate economy level (Dobrowolski et al. 1986) an indispensable theoretical background for programming development was established.

In parallel, through all that time, MIDA system was developed. The system was born in terms of conceptual framework and its basic functional and architectural structure surprisingly early (Borek et al. 1978). This was possible owing to the very strongly problem oriented research and applications being executed in parallel.

But nevertheless a necessity of formulating a basic theory of the field of a DSS application is to be spell out very strongly. The theory enables then for proper elaboration of the DSS architecture and helps in implementation of the system. The system can then be verified and improved from application to application and consequently from version to version.

The basic model of an IS described in (Dobrowolski, Zebrowski, 1988) in the currently stabilized form is sufficiently general and can be applied (and already was applied) in the variety of specific problems going beyond an immediate scope of development programming. It can be also used in practically any process industry. The methodology for the case specific model adaptation was also developed (see (Dobrowolski, Zebrowski, 1988))

A useful theoretical development (described in (Zebrowski, 1988) contributed, through development of the model, to the multiobjective evaluation of industrial structures. It reflects the hierarchy and relation between efficiency and substitution providing a key interface between intermediate economy level of the programming development and the macroeconomic level or the environment of decision making analysis considered. This concept has also wider application potential then original field of MIDA.

The hierarchical decomposition described above, applied to the development programming also may be regarded as a theoretical development which contributed not only to the current state of the MIDA approach and the system. The theoretical developments in spatial allocation and scheduling of investment have the two aspects: they contributed to MIDA development and may be regarded as more generally applicable.

Traditionally from the theoretical point of view, a formal decomposition of the model such as PDA should be considered especially when dimensionality is of concern. Our findings are not in conformity with the traditional approach. We found that practical way leads not through numerical decomposition of large PDA model but through step by step synthesis of smaller models which after being optimized and evaluated are to be integrated into one big model. The following aspects were taken into account. Validation of primary data for PDA model can be efficiently achieved when dealing with smaller models. The same, even to greater extent, considers interpretation of results especially with respect to properties of various technologies, applicability of feedstock, attainability of goals etc.

Therefore it can be summarized that both from theoretical and practical points of view, the real problem is rather on the side of synthesis of large PDA model aimed at generation of efficient development alternatives (alternative development programs) as opposed to mentioned before "traditional" theoretical approach leading through decomposition of primary big model into smaller submodels.

One of important areas of research was, and still remains, the problem of evaluation and selection of development alternatives (development programs) leading to their ranking and selection. The first step was done based on application of SCDAS concept (Dobrowolski, Zebrowski, 1987). The idea of ranking and selection of development alternatives was experimented with on the case of alternative technologies. It was a test example for the idea of application of the proposed approach to the ranking and selection of development programs.

All the identification, theoretical and application activities were accompanied by the software development.

The important synthesizing effect is provided by knowledge and know how gained at the border of all the above activities with focus on implementation of DSS and its software. This can be found in the MIDA architecture and is described in (Dobrowolski, Rys, 1988). Again experience gained in this field goes well beyond a particular DSS. The paper also describes software elements which are parts of a DSS MIDA.

Beside the MIDA development, a variety of other software tools was also developed. It is worth to mention just two examples of packages described shortly in the part 3 of this volume. These are POSTAN — postoptimal analysis package (Dobrowolski et al. in prep.) and PLP or Parametric Programming Package (Golebiowski in prep.). A variety of other software developed could be quoted not only as useful for MIDA but also as generally applicable as every day tools. They provided an important professional upgrade of the team involved.

4 Conclusions and future prospects

When concluding the kind of review of a substantial period of experience in the theory, software and application of a DSS such as MIDA, one should aim at pointing out its general as well as elsewhere applicable aspects.

This can be done from the perspective of the fact that MIDA DSS and MIDA methodology have been developed and applied in the substantial number and variety of cases. More over it was done in a diverse decision as well as cultural environments.

The research program aimed at the development of a DSS for programming development of the chemical industry although was (and is) an open ended one but at the some time was (and is) very strongly problem and application oriented. It was very much supported by the team work organized with participation of high class industrial experts representing both decision making and technical skills. The synergic effect of all the above enumerated elements proved to be decisive for obtaining results reported so far.

Extensive and conceptually wide collaboration with IIASA provided a scientific back up which must not be underestimated.

MIDA does play a double role in the game. It is a permanently improved scientific tool which provides together with already accomplished case studies a unique laboratory for research and application in two related areas: in programming development of the chemical (and process industries) and in development of decision support tools and systems.

MIDA is also a professional DSS package offered as a product on the market. The demand coming from competition exerts a specific kind of pressure on its development and performance.

Now we can naturally involve a problem of learning. In the above mentioned systems role in research as a laboratory tool the aspect of learning should be exposed and considered. This situation is similar to the role of DSS in the process of a decision analysis considered as a process of learning. In MIDA a decision maker is cast in a creative role and interacts with the system in the process of generating efficient alternatives of development. This in fact is a process of learning and a creative thinking. By assuming a creative role of a decision maker we have also assumed a subjective factor to be present in the process, since it represents other side of the creative involvement. Decision maker presides over the process but also must take full responsibility for the effect.

MIDA experience strongly supports profound ideas represented in the book by Stuart and Hubert Dreyfus (1986) with the meaningful title "Mind over machine. The Power of Human Intuition and Expertise in the Era of the Computer".

DSS can only assist a decision maker and experts in their strive to design and select an efficient development alternative. An optimal solution obtained from DSS is to be considered as an important but just a factor in the process.

DSS may also help in training of those who aim at becoming experts. They are bound the climb through the levels proposed by Dreyfuss: from novice, through advanced beginner, competent proficient to the expert.

However when a real application is to be accomplished, the team of experts must represent the highest level of expertise. In some instances, when getting into a new case, an expert may be forced to step down to the level of proficiency, the level of competence may not be acceptable. Then DSS proves to be useful in helping with the quick and efficient upgrade — back to the level of expertise.

The development of MIDA is going to be continued in all respects presented in this paper.

In the field of theory work is foreseen both on mechanisms of development as well as on resource allocation problems, both in space and in time.

Further work in the direction of ranking and selection of development alternatives will, as it is expected, lead to implementation of a new module for MIDA system which would serve as a tool for evaluation of alternatives by a group of experts.

New models are to be developed especially concerning fine chemicals obtained from periodic and batch processing. This would be complementary to MIDA type of DSS covering so called light or fine chemical industry (e.g. colorants, pharmaceuticals) due to its specific technological and marketing properties.

CAD type of approach is envisaged aimed at development of some engineering tools useful both in programming development and design of new technologies.

All these efforts are supposed to be accompanied by methodological developments.

The base and verification for all the activities will be provided by applications.

Acknowledgements

The contribution of industrial experts collaborating in all phases of MIDA development and applications hardly can be overestimated. The gratitude must be expressed to W.Marek, S.Gibinski, J.Wojtania and J.Wilczynski. H.Gorecki contributed substantially in scientific part of development. The collaboration with A.Lewandowski and A.Wierzbicki is to be once again acknowledged.

References

Bellman, R. (1961). Adaptive Control Process — a Guided Tour. Princeton University Press, Princeton, New Jersey.

Dobrowolski, G., H. Gorecki, J. Kopytowski, M. Zebrowski (1982). The Quest for a Concordance Between Technologies and Resources as a Multiobjective Decision Process. Multiobjective and Stochastic Optimization, pp. 463-475, IIASA Collaborative Paper CP-82-512, Laxenburg, Austria.

Dobrowolski, G., Kopytowski J., Lewandowski A., Zebrowski M. (1982). Generating Efficient Alternatives for Development of the Chemical Industry. IIASA Collaborative Paper CP-82-54, Laxenburg, Austria.

Gorecki, H., J. Kopytowski, T. Rys, M. Zebrowski (1984). Multiobjective Procedure for Project Formulation - Design of a Chemical Installation. In: M. Grauer, A.P. Wierzbicki (Eds.) Interactive Decision Analysis - Proc. of Int. Workshop on Interactive Decision Analysis and Interpretative Computer Intelligence Springer Verlag pp. 248-259.

Borek, A.,G. Dobrowolski, M. Zebrowski, (1979). Applications of System Analysis in Management of Growth and Development of the Chemical Industry. CHEM/SEM.8/R.16 Report of Chemical Industry Committee of the United Nations.

Dobrowolski, G., J. Kopytowski, J. Wojtania, M. Zebrowski (1984). Alternative Routes from Fossil Resources to Chemical Feedstock. IIASA Research Report RR-84-19, Laxenburg, Austria.

Dobrowolski, G., M. Zebrowski (1985). Decision Support in Substitution Analysis for IDS - Industrial Development Strategy Exemplified by the Fuel and Feedstock Sector of the Chemical Industry. The Application of DIDAS. Theory, Software and Test Examples for Decision Support Systems, A. Lewandowski and A. Wierzbicki eds., IIASA, Laxenburg, Austria.

Dobrowolski, G., J. Kopytowski, T. Rys, M. Zebrowski (1985). MIDA - Multiobjective Interactive Decision Aid in the Development of the Chemical Industry. Theory, Software and Test Examples for Decision Support Systems, A. Lewandowski and A. Wierzbicki eds.,IIASA, Laxenburg, Austria.

Skocz, M., M. Zebrowski (1986). An Extended Resources Allocation Method in Design of Industrial Development Strategy. *Proceedings of IFAC Symposium on Large Scale Systems, Zurich.*

Zebrowski, M. (1987). Multiobjective Evaluation of Industrial Structures. MIDA Application to the case of the chemical industry. Theory, Software and Testing Examples for Decision Support Systems, A. Lewandowski and A. Wierzbicki eds., WP-87-26, IIASA, Laxenburg, Austria.

Bellman, R. (1961). Adaptive Control Processes - a Guided Tour. Princeton University Press, Princeton, New Jersey.

Kendrick, D. A., A.J. Stoutjestijk (1978). The Planning of Industrial Investment Programs. Vol. 1. A Methodology. Johns Hopkins University Press, Baltimore, M.D., for World Bank Research Publications.

Kopytowski, J., J. Wojtania, M. Zebrowski (1981). Fossil as Key Resources of Hydrocarbons for the Chemical Industry - the Burning Problem of Industrial Development. IIASA Collaborative Paper CP-81-20, Laxenburg, Austria.

Wierzbicki, A.P. (1980). The Use of Reference Objectives in Multicriteria Optimization - Theoretical Implications and Practical Experience. IIASA Working Paper WP-79-66, Laxenburg, Austria.

Dreyfus, H.L., S.E. Dreyfus (1986). Mind over machine. The Power of Human Intuition and Expertise in the Era of the Computer. Second Ed. Bosil Blachwell Ltd., Oxford UK.

Zebrowski, M., G. Dobrowolski, T. Rys, M. Skocz, W. Ziembla (1988). Industrial Structure Optimization: The PDAS Model. Expert Systems for Integrated Development: A Case Study of Shanxi Province The People's Republic of China, Final Report, IIASA, Laxenburg, Austria.

Dobrowolski, G., Rys T. (1981). A Dynamic Model for Development Programming of Production — Distribution Area. Theory and Applications. Materials of School of Economic Systems Simulation, Trzebiatowice (in polish)

Dobrowolski, G., M. Zebrowski (1987). Ranking and Selection of Chemical Technologies Application of SCDAS Concept. Theory, Software and Testing Examples for Decision Support Systems, A. Lewandowski and A. Wierzbicki eds., pp. 233-241, WP-87-26, IIASA, Laxenburg, Austria.

Lewandowski, A., S. Johnson, A. Wierzbicki (1986). A Prototype Selection Committee Decision Analysis Support System SCDAS: Theoretical Background and Computer Implementation. IIASA Working Paper WP-86-27, Laxenburg, Austria.

Zebrowski, M., P. Rejewski (1987). Technological Innovations for Ecologically Sustainable Development, IIASA Working Paper WP-87-17, Laxenburg, Austria.

Rejewski, P., M. Zebrowski (1987). ERIS - Energochemical Regional Integrated System. Paper presented in IIASA Workshop on Integrated Energy Systems, Laxenburg, Austria.

Zebrowski, M. (1987). A Guide to the Integrated Development Programming. Joint Systems Research Department Report for UNIDO Project. Vienna, Austria.

Dobrowolski, G., T. Rys, K. Hajduk, A. Korytowski (in preparation) POSTAN3 - Extended Postoptimal Analysis Package for MINOS. IIASA Collaborative Paper, Laxenburg, Austria.

Golebiowski, A. (in preparation) PLP - A Package for Parametric Programming. IIASA Collaborative Paper, Laxenburg, Austria.

Dobrowolski, G., M. Zebrowski (1989). Basic Model of an Industrial Structure, In: Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems, Vol.331, Springer-Verlag.

Zebrowski, M. (1989). Multiobjective Evaluation of Industrial Structures, In: Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems, Vol.331, Springer-Verlag.

Dobrowolski, G., M. Zebrowski (1989). - Hierarchical Multiobjective Approach to a Programming problem, In: Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems, Vol.331, Springer-Verlag.

Skocz, M., M. Zebrowski, W. Ziembla (1989). Spatial Allocation and Investment Scheduling in the Development Programming, In: Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems, Vol.331, Springer-Verlag.

Dobrowolski, G., T. Rys (1989). Architecture and Functionality of MIDA, In: Aspiration Based Decision Support Systems, A. Lewandowski and A.P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems, Vol.331, Springer-Verlag.

Resource Constrained Project Scheduling Basic Properties

Eugeniusz Nowicki, Czeslaw Smutnicki
Institute of Engineering Cybernetics
Technical University of Wroclaw, Poland

Abstract

The paper deals with the resource constrained project scheduling formulated by Anthonisse et al. (1987). This problem occurs in international research program "International Comparative Study in DSS" coordinated by IIASA, Laxenburg, Austria. In the paper some basic properties and algorithms are presented. Single and multiple criteria approaches are also considered.

1 Mathematical model

The mathematical model of the resource constrained project scheduling problem introduced by Anthonisse et al. (1987) can be formulated as follows. There is a set $\mathbf{R} = \{1, 2, \ldots, m\}$ of resources and there is a set $\mathbf{T} = \{1, 2, \ldots, n\}$ of tasks. The resources are nondivisiable, renuvable, each of them has only one unit. Task preemption is not allowed. For each task $t \in \mathbf{T}$, there is

- a release time a_t and deadline b_t,
- a class $\mathbf{F}_t$ of feasible resource sets, with $\mathbf{F}_t \subset 2^{\mathbf{R}}$ (the class of all subsets of $\mathbf{R}$),
- a due date d_t, an earliness weight v_t and tardiness weight w_t.

For each ordered pair of tasks $(t, u) \in G^0 \subset \mathbf{T} \times \mathbf{T}$, there are limits α_{tu} and β_{tu} ($\alpha_{tu} \leq \beta_{tu}$) on waiting time. It is assumed that graph $(\mathbf{T}, G^0)$ is acyclic and α_{tu}, β_{yu} may be negative. For each task $t \in \mathbf{T}$ and each of its feasible resource sets $F_t \in \mathbf{F}_t$, there is a processing time $p_t(F_t) \geq 0$. Finally there are four weights v_{max}, w_{max}, v_{sum}, w_{sum}. A schedule is a pair (F, S) of functions on $\mathbf{T}$, where $F = (F_1, \ldots, F_n)$ and $S = (S_1, \ldots, S_n)$. In schedule (F, S) task $t \in \mathbf{T}$ is processed by a feasible resource set $F_t \in \mathbf{F}_t$ and has a starting time S_t. The completion time of task $t \in \mathbf{T}$ is defined by $C_t = S_t + p_t(F_t)$, earliness by $E_t = \max\{0, d_t - C_t\}$ and tardiness by $T_t = \max\{0, C_t - d_t\}$. A schedule (F, S) is feasible if it satisfies the following constraints:

(i) $a_t \leq S_t$ and $C_t \leq b_t$ for each $t \in \mathbf{T}$,

(ii) $\alpha_{tu} \leq S_u - C_t \leq \beta_{tu}$ for each $(t, u) \in G^0$,

(iii) for any $t, u \in \mathbf{T}$, if $F_t \cap F_u \neq \emptyset$ then $(C_t \leq S_u$ or $C_u \leq S_t)$.

The problem (P) is to find the schedule (F^*, S^*) minimizing the following criterion

$$Z = v_{max}V_{max} + w_{max}W_{max} + v_{sum}V_{sum} + w_{sum}W_{sum}$$

where

$$V_{max} = \max_{t \in \mathbf{T}} v_t E_t, \qquad W_{max} = \max_{t \in \mathbf{T}} w_t T_t,$$
$$V_{sum} = \sum_{t \in \mathbf{T}} v_t E_t, \qquad W_{sum} = \sum_{t \in \mathbf{T}} w_t T_t.$$

Criteria V_{max}, W_{max}, V_{sum} and W_{sum} represent maximum penalty and total penalty for the earliness and tardiness respectively. If $b_t = \infty$, $t \in T$ and $\beta_{tu} = \infty$, $(t, u) \in G^0$, then the feasible schedule always exists. In the general case, none feasible solution may exists.

The above problem is strongly NP-hard, furthermore, the problem of finding a feasible solution is NP-complete. There are exist no significant results in the literature for the general case of this problem, however there are several results for some special cases, see for example (Slowinski, 1981) and (Talbot, 1986).

The definition of task processing order is now introduced. The task processing order is a pair (F, G), where $F = (F_1, \ldots, F_n)$ and $G \subset \mathbf{T} \times \mathbf{T}$. In (F, G) task $t \in \mathbf{T}$ is processed by a feasible resource set $F_t \in \mathbf{F}_t$, while processing order is represented by the graph $(\mathbf{T}, G)$. Graph $(\mathbf{T}, G)$ must not contains a cycle and is determined by the following condition: for each $t, u \in \mathbf{T}$, if $F_t \cap F_u \neq \emptyset$ then exactly one of the pairs (t, u) or (u, t) belongs to G. From the definition immediately follows that for fixed F, many various G exist. The schedule (F, S), for task processing order (F, G), is feasible if it satisfies the constraints (i), (ii), and

(iv) $C_t \leq S_u$ for each $(t, u) \in G$.

2 General approaches

Since the problem (P) is strongly NP-hard and the expected size of the problem is large, thus only the approximation algorithms seem to be reasonable solution methods. Let $\hat{F}$, $\hat{G}$ and $\hat{S}$ denote the appropriate values generated by an approximation algorithm. The possible approaches are as follows.

(a) Find a solution directly, i.e. find $\hat{F}$ and $\hat{S}$ simultaneously.

(b) Find a solution using a decomposition method. Taking into account the structure of problem (P), the following decomposition method can be recommended:

- two-level decomposition: find $\hat{F}$; for fixed $\hat{F}$ find $\hat{S}$.

- alternative two-level decomposition: find $\hat{F}$ and $\hat{G}$ simultaneously; for fixed $\hat{F}$ and $\hat{G}$ find $\hat{S}$ (this is a polynomial problem and can be solved by linear programming method).
- three-level decomposition: find $\hat{F}$; for fixed $\hat{F}$ find $\hat{G}$; for fixed $\hat{F}$ and $\hat{G}$ find $\hat{S}$ (this is the same linear problem as above).

Note that only in the alternative two-level and three-level decomposition method, certain part of the problem (P) can be solved optimally.

Most of the approximation algorithms are designed in the form of the two-phase algorithm. In the first phase, a starting solution is quickly found, then in the second phase, the solution is improved in an iterative process. The following methods of improvements are recommended on different levels of decomposition:

- changing the resource allocation (upper level),
- interchanging the adjacent tasks (medium level),
- shifting the tasks on the time axis (lower level); this is useful only if the appropriate subproblem on this level is not solved optimally.

From the user's point of view, the decomposition approaches seem to be the most useful in both the manual and the automatic scheduling.

3 Time-optimal allocation

Let us assume that the task processing order $(\hat{F}, \hat{G})$ have been determined by using certain approximation algorithm. Now, the actual problem is how to find the starting times $\hat{S}$. Since $\hat{F}$ is fixed then the task processing times $p_t(\hat{F}_t)$, $t \in \mathbf{T}$ are also fixed. To simplify the notation we use p_t instead of $p_t(\hat{F}_t)$. Note that in the general case, for fixed $\hat{F}$ and $\hat{G}$, the feasible starting times satisfying the condition (i), (ii) and (iv) may not exist.

3.1 A simple case

Let us consider a special case of the time optimal allocation problem, obtained under following assumption

(a) the ready time and deadline constraints are released,

(b) the waiting time constraints are released,

(c) $v_{max} = 0 = w_{max}$,

(d) $(\mathbf{T}, \hat{G})$ contains the path passing through all the vertices $t \in \mathbf{T}$.

From (d), it follows that without loss of generality the tasks can be indexed by integers, accordingly to theirs appearance in that path. The following algorithm determines the starting times for this problem.

Algorithm.

(i) (* initiation *)
$t := 1;\ C_0 := 0;$
$w_j := w_{sum} * w_j,\ v_j := v_{sum} \star v_j$, for all $j = 1, \ldots, n$;

(ii) (* primary starting and completion time of task t *)
$S_t := \max\{C_{t-1}, d_t - p_t\};\ C_t := S_t + p_t;$
if $C_t = d_t$ then go to step (v);

(iii) (* finding first task t^* in the last block B of tasks *)
$t^* = \min\{1 \leq j \leq t :\ C_{i-1} = S_i \text{ for all } i = j, \ldots, t\};$
$B = \{j :\ t^* \leq j \leq t\};$
$\mathbf{T}^0 = \{j :\ j \in B,\ C_j = d_j\}$; $\mathbf{T}^- = \{j :\ j \in B,\ C_j < d_j\}$; $\mathbf{T}^+ = \{j :\ j \in B,$
$C_j > d_j\}$;
(* checking optimality condition *)
if $\sum_{j \in \mathbf{T}^0 \cup \mathbf{T}^-} v_j \geq \sum_{j \in \mathbf{T}^+} w_j$ then go to step (v);

(iv) (* shifting tasks by Δ in the last block *)
$\Delta := \min\{S_{t^*} - C_{t^*-1}, \min_{j \in \mathbf{T}^+}(C_j - d_j)\};$
if $\Delta = 0$ then go to step (v);
$S_j := S_j - \Delta$ for all $j \in B$; $C_j := C_j - \Delta$ for all $j \in B$;
go to step (iii);

(v) (* next task *)
if $t = n$ then stop (* S_t determine optimal starting times $\hat{S}_t$, $t \in \mathbf{T}$ *)
else $t := t + 1$ and goto step (ii).

The main idea of the algorithm is the following. Let us define a block to be maximum subset of task processed continuously without idle time. In the t-th step of the algorithm, the last block is shifted to the left on time axis as long as the goal function value decreases. It means that task starting times S_t are changed but not increased. The above algorithm requires $O(n^3)$ iterations. A slight modification can be made in order to analyze the problem with ready time constraints. It is enough to change in step (ii) the formula $S_t := \max\{C_{t-1}, r_t, d_t - p_t\}$ and in step (iv) the formula

$$\Delta := \min\{S_{t^*} - C_{t^*-1}, \min_{j \in B}(S_j - r_j), \min_{j \in \mathbf{T}^+}(C_j - d_j)\}.$$

However, the extension of this approach to the problem without the assumption (c) is not possible. The appropriate counterexample is shown in Table 1. Evaluating, step by step, the starting times, we obtain the solution $\hat{C}$ given in Table 1. Note that $\hat{C}_1 = 1$, $\hat{C}_2 = 2$ determines the optimal partial solution with respect to the task set $\mathbf{T} = \{1, 2\}$. The solution $\hat{C}$ yields the value of the goal function equal to $\hat{Z} = 25$, while the optimal solution yields $Z^* = 23 < \hat{Z}$.

3.2 The general case

In the general case, the time-optimal allocation of tasks can be found by solving the following linear program (LP) with $3n + 2$ variables E_t, T_t, C_t, $t \in \mathbf{T}$, E, T, and $5n +$

t	p_t	d_t	v_t	w_t	$\hat{C}_t$	C_t^*
1	1	2	3	0	1	2
2	1	2	0	1	2	3
3	1	5	6	0	4	4
4	1	5	150	20	5	5
5	1	5	0	2	6	6

Table 1: Example of problem (P); $v_{max} = 2/3, w_{max} = 5, v_{sum} = w_{sum} = 1$.

$|G^0| + |\hat{G}|$ constraints.

$$\min_{E,T,E_t,T_t,C_t} v_{max}E + w_{max}T + v_{sum} \sum_{t \in \mathbf{T}} v_t E_t + w_{sum} \sum_{t \in \mathbf{T}} w_t T_t$$

under the constraints

$$\begin{aligned}
E &\geq v_t(d_t - C_t), && t \in \mathbf{T}, \\
T &\geq w_t(C_t - d_t), && t \in \mathbf{T}, \\
T_t - E_t &= C_t - d_t, && t \in \mathbf{T}, \\
a_t + p_t &\leq C_t \leq b_t, && t \in \mathbf{T}, \\
\alpha_{tu} &\leq C_u - p_u - C_t \leq \beta_{tu}, && (t,u) \in G^0, \\
C_t &\leq C_u - p_u, && (t,u) \in \hat{G}, \\
E_t, T_t &\geq 0, && t \in \mathbf{T}, \quad E, T \geq 0.
\end{aligned}$$

If a solution of (LP) does not exist, then there exist no feasible starting times $\hat{S}_t$, $t \in \mathbf{T}$, for fixed $\hat{F}$ and $\hat{G}$. Otherwise, the optimal starting times are given by $\hat{S}_t = C_t^* - p_t$, $t \in T$, where C_t^*, $t \in \mathbf{T}$, is the solution of (LP).

In the approximation algorithm based on multilevel decomposition, the problem (LP) is solved many times with slight modification of the input data. Moreover, even for small size of problem (P), the size of problem (LP) is relatively large. Therefore, efficiency of this type of approximation algorithm strongly depends the on efficiency of an algorithm for problem (LP). The further research should be concentrated on:

(a) a quick solution algorithm of problem (LP), e.g. using special structure of constraints; the strongly polynomial algorithm might exist in this case,

(b) a method of improving the existing solution when a slight modification of constraints is made,

(c) an approximation algorithm when the size of problem (LP) is large; the approach presented in section 4.1 can be extended to this case.

4 Single or multiple criteria

Problem (P) considered in section 1 has the single criterion $Z = v_{max}V_{max} + w_{max}W_{max} + v_{sum}V_{sum} + w_{sum}W_{sum}$ defining the global penalty. Problem (P) can also be formulated as a multiple criteria problem (P4) with four criteria $[V_{max}, W_{max}, V_{sum}, W_{sum}]$. Applying typical approach based on a reference point and weighted norm l_1, (Wierzbicki, 1986), the following single criterion is obtained

$$l_1 = \alpha_{max}(V_{max} - V^0_{max}) + \beta_{max}(W_{max} - W^0_{max}) + \alpha_{sum}(V_{sum} - V^0_{sum}) + \beta_{sum}(W_{sum} - W^0_{sum})$$

where $\alpha_{max}, \beta_{max}, \alpha_{sum}, \beta_{sum}$ are certain coefficients and $[V^0_{max}, W^0_{max}, V^0_{sum}, W^0_{sum}]$ is the reference point. Note, that criterion l_1 is equivalent to the criterion Z. Point $[V^*_{max}, W^*_{max}, V^*_{sum}, W^*_{sum}]$ obtained by solving problem (P) with criterion Z is an efficient point in four-criteria problem (P4).

Let us consider the problem (P) as a multiple criteria problem (Pn) with $2n$ criteria $[E_1, \ldots, E_n, T_1, \ldots, T_n]$. Applying reference point approach with norms l_1 and l_∞, we obtained the following ways of goal function scalarization:

$$\begin{aligned} l_1 &= \sum_{t \in \mathbf{T}} v_t(E_t - E^0_t) + \sum_{t \in \mathbf{T}} w_t(T_t - T^0_t), \\ l_\infty &= \max_{t \in \mathbf{T}} \max\{v_t(E_t - E^0_t),\ w_t(T_t - T^0_t)\}, \\ l_{1\infty} &= \max_{t \in \mathbf{T}} \max\{v_t(E_t - E^0_t),\ w_t(T_t - T^0_t)\} + \lambda\Big(\sum_{t \in \mathbf{T}} v_t(E_t - E^0_t) + \sum_{t \in \mathbf{T}} w_t(T_t - T^0_t)\Big), \end{aligned}$$

where v_t, w_t, $t \in \mathbf{T}$ and λ are certain coefficients and $[E^0_1, \ldots, E^0_n, T^0_1, \ldots, T^0_n]$ is the reference point. However, the function $l_{1\infty}$ is not equivalent to the criterion Z. It is obvious that if $[E^*_1, \ldots, E^*_n, T^*_1, \ldots, T^*_n]$ is an efficient point for multiple criteria problem (Pn), then there are exist such $v_t > 0$, $w_t > 0$, $t \in \mathbf{T}$, that $[E^*_1, \ldots, E^*_n, T^*_1, \ldots, T^*_n]$ uniquely minimizes l_∞ over all feasible E_t, T_t, $t \in \mathbf{T}$, see for example (Wierzbicki, 1986). The above property is also true for $l_{1,\infty}$. It can be shown that for problem (Pn), there exist efficient points that may not be obtain by solving problem (P) with criterion Z (i.e. there exist no the appropriate weights v_t, w_t, $t \in \mathbf{T}$, v_{max}, w_{max}, v_{sum}, w_{sum}).

Some scheduling problems with criterion $F = \max_{t \in \mathbf{T}} \max\{g_t(E_t), f_t(T_t)\}$, where g_t, f_t are nondecreasing functions were considered. Note, that criterion l_∞ is a special case of F. A single machine scheduling problem with criterion F and some additional assumptions regarding F was examined by Sidney (1977). For the flow-shop problem, the similar results were obtained by Achutchan et al. (1981). Without any additional assumptions, the flow-shop problem was solved by Grabowski and Smutnicki (1986); this approach can be used for problem (P4).

References

Achutchan, N. R., Grabowski, J., Sidney, J. B. (1981). Optimal Flow Shop Scheduling with Earliness and Tardiness Penalties. *OPRSEARCH*, 4.

Anthonisse, J. M., van Hee, K. M., Lenstra, J. K. (1987). Resource-constrained project scheduling: an international exercise in DSS development. IIASA Report. Laxenburg, Austria.

Grabowski, J., Smutnicki, C. (1986). Sequencing Problem with Minimax Penalty Function (Polish). Archiwum Automatyki i Telemechaniki, 1–2, pp. 21–37.

Sidney, J. B. (1977). Optimal Single-machine Scheduling with Earliness and Tardiness Penalties. *Operations Research*, 25, pp. 62–69.

Slowinski, R. (1981). Multiobjective Network Scheduling with Efficient Use of Renewable and Nonrenewable Resource. *European Journal of Operational Research*, 7, pp. 265–273.

Talbot, B. F. (1986). Resource-Constrained Project Scheduling with Time-Resource Tradeoff. The Nonpreeptive Case. *Management Science*, 28, pp. 1197–1210.

Wierzbicki, A. P. (1986). On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8, pp. 73–87.

Computer Aided Decision Support for Planning and Management of Research and Development

Peter Parizek and Thomas Vasko
Institute for Applied Cybernetics
Bratislava, Czechoslovakia

1 Introduction

Research and development is planned in many countries on a national level and thus managed centrally by some (usually governmental) authority. The planning process requires supporting information and methodology background because the final R&D plan must take into account many factors like:

- structure of the national economy and its goals for the desired period
- R&D potential of the research community
- expected trends in technology and economy
- international scientific cooperation and connections between research groups and institutes
- amount of money for R&D for this period
- other possible consequences in economy, technology and society

From this point of view the problem of planning of R&D is a multiattribute (or multicriteria) problem. More people are usually responsible for the final decision and then the problem can rise to a group decision making problem.

The aim of this paper is to describe the decision support system MDS, which is being developed to support the management of research and development on the national level. The problem is described from the point of view of necessary information support. We also introduce possible methodology for solving the problem and a brief description of computer implementation on a XT/AT compatible personal computer.

2 Problem description

Research and development (R&D) on national level is usually managed in longer time periods (i.e. — five year plans). Assume, the aim of R&D is described in global programs $P(1), \ldots, P(p)$, which may be thought of as expected levels of science and technology in selected areas of national economy. The description of programs is usually verbal, each program may be subdivided into particular goals $G(1), \ldots, G(g)$, with possible quantification of expected results. The goals may be reached by possible alternative projects $A(1), \ldots, A(a)$, which are objects of evaluation. The aim of R&D planning is to propose those projects for the plan, which in common reach the goals, programs and do not exceed given limitations. This procedure is usually provided by managers, experts and analysts from several areas of science, technology and economy. This group is usually supported by several groups of supporting staff which is responsible mainly for information support for committee members. The duty of management committee is to:

1. set up the programs and their goals of R&D for the next planning period
2. define and evaluate criteria for project selection
3. collect proposals for research projects from research community and/or from government
4. make the decision about the R&D plan — choose the appropriate set of projects according to criteria, fulfilling the constraints of money and reaching the expected goals
5. check running projects periodically — stop non effective projects and add new ones from a permanent stock of new proposals

To fulfill these duties management committee must be provided by an immense information support covering the questions of reached levels of technology and their prognosis for the planned period as soon as the questions of economy and policy. Each of these five phases can be solved separately being in its nature a part of a multicriteria decision procedure. We shall describe and formalize these phases.

2.1 Seting up the programs and goals

The programs and goals of R&D must be set up long enough before the start of the planning period, so that there is enough time to provide phases 2, 3 and 4, which require much more information support and a more complicated procedure. The description of global programs usually cannot be formalized. The collection of goals supporting one global program can be thought of as a research program. If the goals require a certain level of technology which is measurable by any physical units as parameters, then these parameters should be included into the set of criteria for the appropriate procedure in the phase 2.

2.2 Defining the criteria

Criteria for selection of projects may be of different nature — quantitative or qualitative, focusing the aspects of technology economy, ecology, sociology, level of international cooperation, level of cooperation between projects in the same research program and in different programs, expected profits, etc. As the goals may differ essentially (i.e. high technology in machinery should have different approach from, say, biotechnology), the defining of criteria must be done for each goal.

Criteria must be set up to enable composing results from separate goals in programs. The connections between goals and programs with help of rule-based system must be stressed here.

The group of decision makers has to come to a unique set of criteria and rules for the process. The criteria are first defined verbaly, then the bounds, measuring units, importance and/or aspiration levels must be set up. This phase (and all subsequent phases too) should not be performed without a computer aided decision support system.

2.3 Collecting of projects

This phase could be thought of as only an administrative procedure. The main goal of this step is rejecting of those project proposals, which are not acceptable. The project proposals should be checked according to criteria with binary evaluation (YES/NO) and principal importance which can be defined. Such a procedure speeds up the whole decision process although it can be fully performed by the supporting staff. As the result of this phase the system is provided by a set of acceptable projects which moreover fulfills all principal restrictions.

2.4 Decision about R & D plan

This is the final and most important phase of the whole decision procedure. It is to be performed in three steps according to the structure of the problem and defined sets of criteria and alternative projects. Relevant steps in this phase require decision in

- composing chosen projects to research programs and goals
- comparing the proposed projects and choosing the best
- getting together the whole R&D plan

This is in general a hierarchical problem of multicriteria evaluation of finite set of alternatives, although there are differences between the methodology which can be used in particular cases. To cover all consequences, limitations and exceptions a rules-based system with knowledge base about given problem should be present. There is one essential limit for the whole planning process:

the sum of all money requirements cannot exceed M

The procedure of comparing the proposed projects in the particular goal of the program evokes problems of methodology. The final ranking of projects for a particular goal has to provide an evaluation of each project in such a way that the values of projects in different goals might be compared in later steps. On the other hand, the nature of the decision situation in different areas of science and technology requires various methods of evaluation (i.e. interval scale methods, pairwise comparison methods, Saaty methods, aspiration level methods, etc.), which must be modified to give the best result in evaluation of projects.

The final step of the decision process is to select projects according to the final ranking and check the distribution of given amount of money and other limitations for programs proportionally with respect to the selection inside the program.

2.5 Checking of running projects

Running projects are to be checked according the effectiveness during the planning period. The result of these procedure can be stopping of non effective projects and thus providing an additional amount of money for new projects, which can be added during the planning period. This requires a huge information support to enable the comparison of running projects with the latest trends in the 'other' world. Such a policy requires a permanent dialogue between the management committee and the research community and submitting of new suitable project proposals.

3 Information support

The management committee must be provided by an adequate information support concerning all running and prepared projects. The information support should be the duty of the supporting staff and it may be done in a computerized form as a database on the same computer as the decision support system itself. The main problem of information support for management activities described above is an efficient information transfer between a database used for the standard information support in other management activities and the decision support system. There are several ways how to solve this problem. A DSS system may use its own data management system with no import or export data facilities to other data structures. The latest version of SCDAS (Fotr and Pisek, 1986) is an example of such a stand-alone system. From the point of view of a decision maker it may be important to have the same interface to both — information in a database and the decision support system or to have the possibility to access data in the database from the DSS environment.

Another problem is the access to information databases with some factographic data on mainfraims. The online connection to such facilities is an important and necessary need for the management committee and its supporting staff. In the decision support system described later we used a standard database environment to maintain the data.

4 Computer implementation — system MDS

MDS is an all-purpose system (environment) for multi-criterial evaluation of discrete alternatives. Its all-purpose nature is characterized by the following properties:

- it enables a simultaneous solution of several independent decision-making problems which are freely organized into groups
- the system is open towards the used methods, which means that an unlimited number may be included in the system

- the individual algorithms — methods may be implemented in any programming language and they are not a direct part of the shell of the system
- in the simplest case the system may serve as an overview database system of alternatives and their parameters

4.1 Characteristics of the MDS system

The system is hierarchically divided into two levels:

a. – level of programs $P(1), \dots, P(p)$
b. – level of goals $G(1), \dots, G(g)$

Under the term "program" we understand complex areas of evaluation, e.g. program of electronisation, program of development of the food industry, program of construction on a certain building site etc.

Every program $P(i)$ may incorporate any number of goals. Under the term "goal" we understand a concrete evaluation problem to be solved to reach the goal, means to find the appropriate alternative from the proposed set. The goal is characterized by attributes (criteria) $C(1), \dots, C(c)$, eventually rules $R(1), \dots, R(r)$, whereas alternatives $A(1), \dots, A(a)$ are the objects of evaluation. Further characteristics of the goal may eventually be created in the system with the help of special methods.

Attributes together with rules describe the characteristic of the goal. The selection of appropriate attributes is made by experts $E(1), \dots, E(e)$ or by the moderator of the decision process. From the point of view of the decision-making process we divide the attributes into qualitative and quantitative depending on their values being objectively measurable (which means they may be numerically expressed and input beforehand, e.g. technical parameters, numbers, weight etc.) or being the object of evaluation by experts.

By the means of rules experts express general characteristics of the goal, which can not be expressed by the values of attributes or where this presents considerable difficulties. Rules serve to define borders, limitations, eventually relations between attributes. Rules are of the IF–THEN type, where in the conditional and active part attributes are present. Every rule has an allocated weight, which influences the final evaluation of alternatives. The rules are defined by the experts together or are input by the author of the program.

Every goal may be reached by any number of alternatives. The aim of the MDS is to specify the optimal ranking of alternatives with the help of the knowledge expressed in attributes, in the weights of the particular rules and in the alternatives employing methods of the multi-criterial evaluation.

Experts from the given field participate in the definition of the decision-making task and its solution. Their responsibility is to choose the attributes and general rules for the given goal, to select the weighs of the attributes and to determine the qualitative attributes of every alternative.

The management of the whole system is a responsibility of the so-called manager of the program, who is the only person who may effect changes of parameters during the whole activity of the system. Some of his tasks are to define the following:

- goals of the program, i.e. define the structure of the decision-making problem

- weights of the experts
- attributes (after consulting the experts)
- rules (after consulting the experts)
- alternatives (after consulting the experts)
- values of the quantitative attributes, i.e. those which may be objectively measured or set up
- method for attribute weight or aspiration/reservation level definition
- method for value input for qualitative attributes of the alternatives
- method for aggregation of all values from experts
- method for evaluation of disagreement indicator
- method for final ranking of alternative evaluation

The procedure of the input of above mentioned parameters depends on the chosen methodology for evaluation.

4.2 Description of system usage

We may divide the usage of MDS into four stages:

1st stage, preparation, includes the following activities:

- definition of the problem, i.e. the goal of the multi-criterial evaluation of the alternatives. If the new goal is already a part of an existing program, it will be added to it, otherwise a new program with this goal will be set up
- selection of experts for the given goal, eventually an allocation of weights to the experts
- definition of attributes and general rules for the given goal by the experts together with the manager of the goal
- selection of appropriate methods for the definition of attribute weigh or aspiration/reservation level, for the assessment of qualitative attributes of alternatives, for the aggregation of all individual expert values and the method of final evaluation of alternatives
- selection of alternatives feasible for evaluation
- input of quantitative attribute values

The number of experts, attributes and alternatives may be changed by the author of the program at any stage. By starting the generation procedure from the communication module MDSGENER (see below) will the MDS effect a restructuring of all databases according to the most recent changes, whereas unchanged data remain preserved. In some cases this step requires to return to previous stages of the system.

2nd stage — assessment of attribute characteristics by experts
Every expert evaluates characteristics of attributes in MDSDIAL module employing the method which was chosen by the manager of the task. Afterwards the system evaluates the disagreement indicator. After an eventual correction of the individual assessments MDS aggregates the characteristics into the final evaluation employing the chosen method.

3rd stage — evaluation of qualitative attributes of alternatives
This activity is performed by experts in a way similar to the assessment of attribute characteristics in the 2nd stage. After the aggregation of qualitative attributes the results together with quantitative attributes form the final individual characteristic of every alternative. A general characteristic (impression) of the expected alternative may be defined by rules and/or in the so-called basic alternative.

4th stage — final evaluation of alternatives employing the chosen method
This stage comprises evaluation of rules and presentation of results. If there are some rules connections between goals or programs defined in knowledge base of the system an appropriate complex results are derived by the system. The system supports final evaluation output in text and graphic form to the screen as well as to the printer.

4.3 PC implementation

The MDS system is a heterogeneous system, as far as the programming tools are concerned, which means that various parts may be programmed in different programming languages. The unifying environment is formed by the system databases of dBASE III+ format and by two communication modules programmed under the database system CLIPPER. All further modules, programmed in different programming languages, are called from the communication modules with the use of the command RUN which is a part of the CLIPPER language. The cooperation of the various modules takes place through the databases. The methods of reading and writing these databases are known, as their structure is described in manuals.

There were several reasons to form such a heterogeneous system, built from databases and independent routines:

- an effort to process data with the help of a professional database system, e.g. CLIPPER, so that it would not be necessary to undergo the time-consuming programming of subroutines in some of the common programming languages to obtain functions such as editing of data, seeking for data or data retrieval. This approach ensures some degree of comfort in the system with the possibility of archivation, copying, import from text files, eventually employing data from other professional software packages which recognize the dBASE III+ data structure

- an aim to create a communication environment with a modular concept to enable the use of various subroutines written in any programming language. These subroutines may include the algorithms or supporting subroutines, e.g. for graphic output.

- a further required property of the system was a possibility of increasing the number of employed methods

By complying these requirements an all-purpose system was created which actually forms an environment for the tasks of multi-criterial selection of discrete alternatives.

The system is implemented on the IBM PC-XT/AT with 640KB RAM. The communication modules are able to operate independently on the used video controller, the modules for graphic output depend on the used software package, which must respect the configuration of the video controller. In regard of the size of the whole system and in order to increase operation speed the use of a hard disk is recommended. The maximum number of attributes and experts is 1020. The number of programs, goals, rules and alternatives is limited by the maximum database size in the CLIPPER database system, i.e. approximately one milliard records. From the above mentioned it is clear, that the size and number of solved tasks is practically limited only by the disk memory size of the computer.

5 Summary

Such a broad-concept, open system with a modular structure offers a wide variety of practical use. For specific applications it is possible to create versions which are exactly "tailored to suit the needs" for one or several problems in a short span of time but nevertheless with all comfort required for software products of this kind, which also has a big importance from the commercial point of view. A combination of this system with an real expert system is expected in near future which will further improve the quality of this system and will enable a wider application in the field of selection of discrete alternatives. At present a library of standard methods is being created and a knowledge base for the selection of appropriate methods for MDS applications is being built.

References

Fotr, J. and Pisek, M. (1986). Exact Methods for Economical Decisions, Academia, Prague.

Lewandowski, A., Johnson, S. and Wierzbicki A.P. (1986). A prototype selection committee decision analysis and support system, SCDAS: theoretical background and computer implementation. Working Paper WP-86-27, IIASA, Laxenburg, Austria.

Lewandowski, A. and Wierzbicki, A.P. (1987). Interactive Decision Support Systems — the Case of Discrete Alternatives for Committee Decision Making. Working Paper WP-87-38, IIASA, Laxenburg, Austria.

Lewandowski, A. and Wierzbicki, A.P. (1988). Aspiration Based Decision Analysis and Support, Part I: Theoretical and Methodological Backgrounds. Working Paper WP-88-03, IIASA, Laxenburg, Austria.

Lightyear, (1984). The Decision Support Software, Lightyear, Inc.

Parizek, P. and Vasko, T. (1987). MDS — an Interactive System for Multicriteria Evaluation of Alternatives — a Prototype Version, (short announcement). Proceedings from the International Workshop on Methodology and Software for Interactive Decision Support, Albena, Bulgaria (in preparation).

Parizek, P. (1988). Planning and Management of Research and Development as a Multicriteria Decision Problem. A contribution submitted for the VIIIth International MCDM Conference, Manchester, UK (to be published in the Conference Proceedings).

Sprague, R.H. and Carlson, C., eds. (1982). Building Effective Decision Support Systems, Prentice Hall, Inc.

Multiobjective Investment Scheduling Problem

Tomasz Ryś, Wiesław Ziembla
Joint Systems Research Department
of Institute for Control and Systems Engineering,
Academy of Mining and Metallurgy, Cracow and of
Industrial Chemistry Research Institute, Warsaw

Abstract

In the field of industrial development a problem of investment scheduling often arises. This problem can be treated as an autonomous activity and it comprises a search for optimal (or suboptimal) construction schedules of industrial complexes. The industrial complex means here a set of plants (technologies) which are strongly technologically interdependent and share a common infrastructure.

In the paper, formulation of the investment scheduling problem is presented as a multiobjective task. The following objectives are considered: the completion time, the production profit and investment. Additionally total penalty for lateness is also considered in order to minimize construction delay. Due to technological and market conditions specific predecessorship relation between complexes are also taken into account.

A procedure for solving the above problem and an example is also presented.

1 Introduction

Industrial development and its programming can be viewed as a process of changing production structure by means of investment over the course of time. Due to its complexity and dynamic character when considered as a decision process it is to be decomposed in terms of space and time allocation of investment. The essence of development programming process as well as its decomposition and hierarchy [1] were presented at this conference by J. Kopytowski and M. Żebrowski and also published (J. Kopytowski and M. Żebrowski 1989).

In this paper emphasis is given to the problem of time allocation of investment considered as a decision problem of investment scheduling. In general this problem is very common and can be considered autonomously from other procedures and tasks in the management of industrial development.

[1]The methodology for development programming and a DSS are called MIDA, i.e. Multiobjective Interactive Decision Aid

2 Investment scheduling problem

Problem of investment scheduling can be viewed as a problem of setting start-up dates of plants selected for investment program. It is to be accomplished in such a way so as to maximize profit from investment with respect to demand distributed over a given time horizon. This is also to be done under constraints on availability of capital (distributed over time) and other resources such as energy or construction potential. In the quite common situation such as in the case of a developing country the above problem has also more general dimension. This is necessity stemming from imperative of minimization of import and maximization of export in order to improve trade balance and acquire means for repayment of investment loans. Even the above simple verbal description shows that the investment scheduling is in fact a multiobjective optimization decision problem.

In order to formulate a mathematical model we introduce a basic identity namely industrial complex or shortly complex. Complex is a set of technologies (installations) which are closely technologically interrelated and generally utilize a common infrastructure. In a specific case complex may consist of only one installation. Each complex constitutes an investment task which is subject to scheduling. For a set of complexes there is defined predecessorship relation of a partial order type.

The relation may be imposed by indispensable material flows between complexes as well as demand distribution or none salability of certain semiproducts (which therefore have to be consumed by other complexes). In this case, however on the contrary to the common scheduling theory, the constraints are imposed trough the completion dates.

An important time type of parameters, which can act as constraints or be involved in penalty criteria, are so called relase and due dates. They reflect necessary time coordination with other industrial branches. For example a release date, may be imposed by date of availability of electricity from a power station or a due date may be the starting date of the production for which a rubber must be supplied from a rubber plant or otherwise it would have to be imported. The value of this import represents monetary equivalent of the relaxation of the constraints imposed by release and/or due dates.

The necessity of this kind of a coordination is present in any type of economy but is more critical, due to the scarcity of resources and lack of convertible currency in developing countries.

In the process of investment scheduling basic constraints come from availability of resources. In the model presented here there are considered two basic constraints of this type: investment level and construction potential. First one is of the nonrenewable type (during the construction period) second is a renewable one. Both types of constraints are made additive since they are expressed in monetary terms. The model allows however for differentiating various categories of construction potential (which than could be expressed in terms of man-shifts or construction machinery potential etc should this factors be critical). Differentiation of construction cost may come also from the financing mode. In case such as "turn the key on" type of plant contracting a construction period as a rule is shorter but plant is more expensive. Another situation arises from changing debt/equity ratio when a higher ratio may rise availability of capital on the expense of a higher loan. Contribution from all such possibilities should be optimized with respect to the whole schedule while global investment can be minimized and/or subject to constraints.

3 Mathematical model formulation

Now we may present mathematical formulation of investment scheduling problem. The symbols are categorized as follows :

$k, l \in K$ industrial complexes,

$< k, l > \in R$ predecessorship relation,

d_k due dates (or dead line) for k-th industrial complex,

e_k release time for k-th industrial complex,

p_k complex construction duration k,

s_k, c_k starting time of construction and start-up (starting time of operation) for k-th complex,

$t \in [T_0, T]$ time unit (scheduling level), where T scheduling horizon,

r_{kt} requirement on investment capital (or other resources) for k-th complex in t-th time of its construction,

R_t availability of investment capital (or other resources) in t-th time of scheduling period,

f_k profit for k-th complex, for simplicity :

$$f_k(t) = \begin{cases} 0 & \text{if } t \leq s_k + p_k - 1 \\ const & \text{if } t \geq s_k + p_k \end{cases}$$

but generally it can be nondecreasing function equal zero for $t \leq s_k + p_k - 1$,

v_k penalty function for lateness (or value of equivalent import)

$$v_k(t) = \begin{cases} 0 & \text{if } s_k \leq d_k \\ v_k t & \text{if } s_k > d_k \quad t_k = s_k - d_k \end{cases}$$

Next, constraints can be specified:

- **predecessorship constraints**

$$s_k + p_k \leq s_l + p_l \quad < k, l > \in R \tag{1}$$

- **dead line constraints**

$$s_k + p_k \leq d_k \quad k \in K \tag{2}$$

- **release date constraints**

$$s_k \geq e_k \quad k \in K \tag{3}$$

- **investment (resource) time balance**

$$\sum_{k\in K} g_k(t) \;\leq\; R_t, \quad t \in [T_0, T] \tag{4}$$

where

$$g_k(t) = \begin{cases} 0 & \text{if } t < s_k \text{ or } t > s_k + p_k \\ r_{kt} & \text{if } t \in [s_k, s_k + p_k] \end{cases}$$

Objective can be formulated as follows:

- **Maximum total profit**

$$T_p = \sum_{k\in K} f_k(T - s_k - p_k) \tag{5}$$

$$\max TP = \min \sum_{k\in K} f_k s_k \tag{6}$$

- **Minimum completion time**

$$CT = \max(s_k + p_k) \tag{7}$$

- **Minimum sum of penalty for lateness** in completion time with respect to due date

$$IM = \min \sum_{k\in K} v_k \max[0, s_k - d_k] \tag{8}$$

The above criterion is useful and should be applied when relaxation of dead lines is either acceptable or indispensable due to their violation (infeasibility of a problem).

According to consideration presented in section 2 model can be extended by using the criterion on global investment minimization:

$$IN = \min \sum_{t=T_0}^{T} \sum_{k\in K} g_k(t) \tag{9}$$

$$g_k(t) = \begin{cases} 0 & \text{if } t < s_k \text{ or } t > s_k + p_k \\ r_{jkt} & \text{if } t \in [s_k, s_k + p_k] \\ & \text{and } x_{jk} \;=\; 1 \end{cases}$$

Binary variable x_{jk} is used for assignment industrial complex to the mode of its completion:

$$x_{jk} = \begin{cases} 1 & \text{complex } k \text{ is completed in the way } j \\ 0 & \text{otherwise} \end{cases}$$

4 Problem solution

The problem formulated above is a problem of mixed integer linear programming type. Unfortunately this kind of problems even for a single criterion belong to NP-complete class due to their computational complexity. The accurate method for solving such problems are based on implicit enumeration (e.g. Talbot and Paterson 1978) In the case of presented here investment scheduling in order to obtain practically applicable decision support, a methodology based on a combination of heuristic and approximate algorithms was developed. The results so far confirm practical applicability of the approach. This comes merely from the fact that most important aspect of decision making with respect to scheduling is necessity of evaluation and ranking of various schedules rather than costly and laborious enumeration of every alternative.

Below are given heuristic rules based on identification and practical experience when dealing with single criteria problems. They can be presented as follows:

H1 **Maximum profit** : this corresponds to perequisite to sequencing complexes in the descending order of their respective individual profits;

H2 **Minimum construction time:** this corresponds to perequisite to sequencing complexes according to:

- descending order of the investment capital of complexes
- descending order of complexes' maximum values of investment expenditure in any time during their completion periods;

H3 **Maximum ratio of profit over investment:** this corresponds to the perequisite of getting best attainable adjustment of resources consumed to the resources available - assuring maximum profitability. It can be fulfilled by descending order of profit over investment ratios. In this rule both ways of ordering enumerated in the formula H2 can be utilized.

H4 **Minimum total penalty for lateness (or for the import substituting domestic production):** This corresponds to the perequisite of ordering complexes along descending values of expression: $v_k(T - d_k)$.

H5 **Maximum profit and minimum total penalty for lateness** : This corresponds to the perequisite of ordering complexes along descending values of expression: $f_k(T - p_k) + v_k(T - d_k)$.

Sequences resulting from the above heuristic rules are to be adjusted according to predecessorship relations. Due to the highest priority of this relation each sequence must bent to it - should any conflict arise. The computer implementation of the model does it automatically using for regrouping the same heuristic rules.

As a result of sequencing according to the rules H1 - H5 and their combination the investment schedule obtained, provides de facto construction priorities for the considered complexes. Additionally decision maker has also authority for (interactively) imposing own preferences. This gives him possibility to modify priorities resulting from the above rules.

Next, we may consider a multiobjective case. From the variety of approaches an ϵ - constrained (or bounded objective) method will be discussed. Corresponding to such a case the optimization problem will be in a form:

$$\min_{s_k, k\in K} \sum_{k\in K} f_k s_k \tag{10}$$

s.t.

$$\sum_{k\in K} v_k \max(0, s_k - d_k) \leq M^0 \tag{11}$$

$$\max_{k\in K}(s_k + p_k) \leq CT^0 \tag{12}$$

$$s_k + p_k \leq s_l + p_l \quad <k,l>\in R \tag{13}$$

$$\sum_{k\in K} g_k(t) \leq R_t \tag{14}$$

For solving the above problem we have selected dual programming (Fisher 1976) combined with subgradient search strategy (Held and all 1974).

Let constraints (12) and (13) define a set of feasible solutions denoted by D . Next, let us introduce dual variable ρ, λ_t related to constraints (11) and (14) and define Lagrange'a function:

$$\begin{aligned} L(s,\rho,\lambda) &= \sum_{k\in K} f_k s_k + \rho \sum_{k\in K} v_k \max(0, s_k - d_k) + \sum_{t=t_0}^{CT^0} \lambda_t \sum_{k\in K} g_k(t) = \\ &= \sum_{k\in K} \{ f_k s_k + \rho v_k \max(0, s_k - d_k) + \sum_{T=s_k+1}^{s_k+p_k} \lambda_k r_{k,t-s_k} \} \end{aligned} \tag{15}$$

Based on the above formulation λ i ρ can be interpreted as cost payable for utilization of additional resources utilization and as a penalty for total weighted lateness. Now we can define so called Lagrange'a problem $W(s,\lambda,\rho)$:

$$W(s,\lambda,\rho) = \min_{\{s_k\}\in D, k\in K} L(s,\lambda,\rho) - \rho I M^0 - \sum_{t=T^0}^{CT^0} R_t \lambda_t \tag{16}$$

The definition of $W(s,\lambda,\rho)$ simply implies, that $W(s,\lambda,\rho)$ is a lower bound for objective function (10) for any $\lambda_t \geq 0$ i $\rho \geq 0$. The best lower bound can be obtained for $\max_{\lambda\geq 0, \rho\geq 0} W(s,\lambda,\rho)$, and that leads to two level iterative algorithm to solve problem (10) - (14) as shown in figure 1.

The lower level algorithm is of a recurrent type and is based on additive formulation of Lagrange'a function, where each element depends only on a simple complex start-up variable. The upper level is a subgradient search algorithm, generating sequence of values λ and ρ variables in each iteration. Detailed description of the algorithm is given in papers (Ziembla 1987, Skocz and all 1987). The first upper bound value in these formulas is calculated using heuristic rules, next ones are improved by actually best sequence generated by nondecreasing order of calculated start-up dates for considered complexes.

However,it has to be pointed out that the above procedure (Held and all 1974) is approximate and no formal proof of its convergence exists. It allows only for enumeration of a lower bound of objective functions. To obtain accurate solution, presented method has to be combined with algorithms of branch and bound. This approach is currently being tested for release of practically applicable software package.

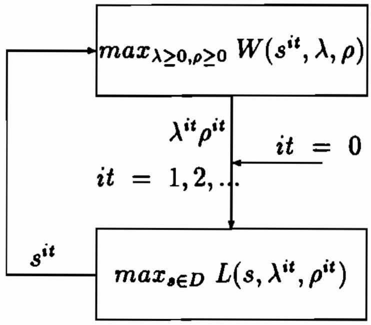


Figure 1: Two level iterative algorithm.

5 Practical experience

Investment scheduling has been implemented as an autonomous module of the mentioned in the introduction MIDA system. But as it was pointed out it can be applied separately. Scheduling was applied in various development projects (Kopytowski and Żebrowski 1989). To better illustrate the case, below will be presented briefly an example based on a development case for the petrochemical industry.

The results are shown for two alternatives. First (see fig. 2) is the case of uneven distribution of capital while the second case corresponds to the even capital distribution. These two alternatives for a global constraint lead to evidently different investment schedules as seen on fig. 3. For the sake of simplicity of presentation only one criterion namely maximum total profit was considered.

6 Summary and conclusions

The complex, theoretically and numerically, problem of investment scheduling proved to be practically solvably by applying simple heuristic rules combined with multiobjective optimization. This combination yielded practical decision support tool which could be incorporated in MIDA system and also can be applied separately. An important aspect of this application is that decision maker can interactively influence the process of scheduling by learning about impacts and effects of his preferences.

Method helps not only to produce a schedule for investment program but is also helpful in verifying the selection of projects (complexes) with respect to the attainability of the development program. This may even lead to structural changes as well as to revision of the development strategies. On the other hand effects of relaxation of investment as well as time constraints enable for finding out their feasible modifications or give ground for

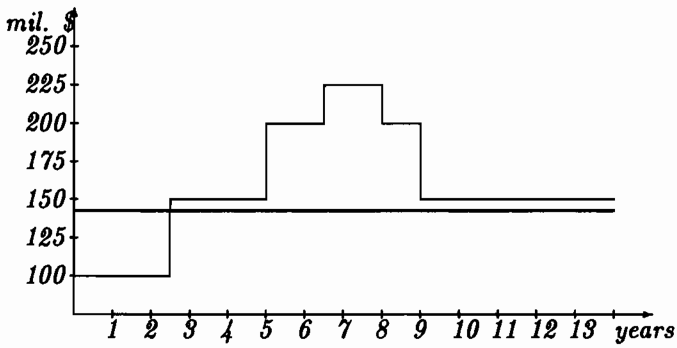


Figure 2: Availability of investment capital - even and uneven distribution

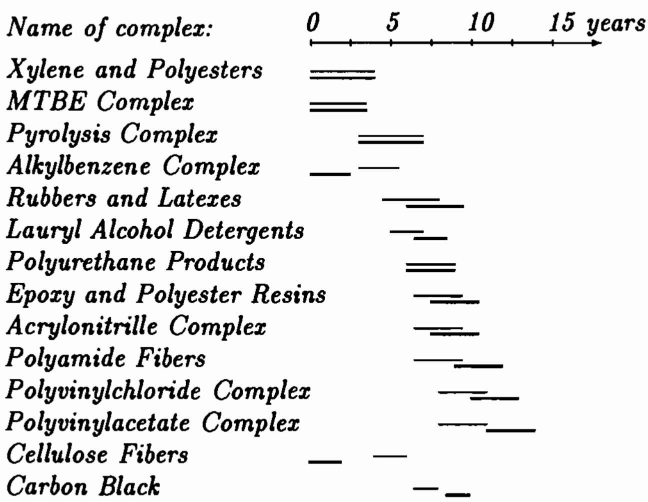


Figure 3: Gantt chart for two cases of capital distribution.

potential renegotiation of bank or government policies.

Acknowledgements

Authors would like to express their gratitude to dr M. Żebrowski for contribution and close collaboration on the primary version of the model. His expertize in development programming cannot be overestimated.

References

Fisher M.L. (1976) A Dual Algorithm for the One-Machine Scheduling Problems, *Math. Progr.* Vol 11, pp. 229-251.

Held M., P.Wolfe and H.Crowder (1974) Validation of Subgradient Optimization, *Math. Progr.* Vol 16, pp. 62-88.

Kopytowski J. and M. Żebrowski (1989) MIDA : Experience in Theory, Software and Application of DSS in the Chemical Industry. In: Aspiration Based Decision Support Systems, Lewandowski A. and A.P.Wierzbicki eds., pp. 271-286, Springer-Verlag.

Skocz M., M. Żebrowski, W. Ziembla (1987) A Method for Design of Industry Development Strategy. In: Preprints of X IFAC Congress in Munich 1987.

Talbot F.B. and J.H. Paterson (1978) An Efficient Integer Programming Algorithm with Network Cuts for Solving Resource Constrained Scheduling Problem *Management Sci.* Vol 24, No 11.

Ziembla W. (1987) Scheduling Problems with Interoperation Constraints, PhD Thesis (in polish), Academy of Mining and Metallurgy, Cracow.

Resource Assignment in a DSS for Project Scheduling

P. Serafini, M. G. Speranza
Dip. di Matematica e Informatica
via Zanon, 6 – I–33100 Udine, Italy

1 Introduction

Scheduling problems represent a class to which a considerable amount of research effort has been devoted. The reason for such an interest is that this class encompasses many and very different problems, in terms of characteristics, application field and solution techniques.

For historical reasons, inside the scheduling class the project scheduling problems have been dealt with separately, usually through network analysis (see, for a classic review and for a categorization of project scheduling Davis, 1973 and, for more recent results, Davis and Patterson, 1975; Olaguibel and Goerlich, 1988 and Patterson, 1984). As far as the other scheduling problems are concerned, they have been studied through combinatorial optimization techniques (see Conway et al., 1967 for a historical relation with queueing models) and have been more recently classified from a computational complexity point of view (see French, 1982; Lenstra and Rinnooy Kan, 1985; Lawler et al., 1982 and Rinnooy Kan, 1976). In (Bellman et al., 1982) an attempt is made to link the two classes of scheduling problems. As a matter of fact, there are overlappings between the two classes. For instance, the problem of minimizing the total project duration under fixed resource constraints is a generalization of the job shop problem.

In this paper we deal with a resource constrained project scheduling which is strongly related to the job shop problem. A number of tasks have to be processed in order to minimize the makespan or project duration. Precedence constraints can be established among the tasks. The resources are renewable and discrete, in particular only one resource unit is available at each time instant. Each task can be processed in a number of ways, where each alternative requires a set of resources and a processing time. Moreover release dates and deadlines are associated with each task.

It is clear that this is a very complex problem from a computational point of view. It is expected to use heuristics for its solution and so a human interaction may help in deciding the proper solution strategy.

Moreover, it should be kept in mind that scheduling problems exhibit features which are hard to be formalized. These concern for instance the presence of different criteria to be optimized for the selection of an "optimal" schedule. This reason makes a human interaction with the solution procedure almost compulsory.

Therefore we think that a Decision Support System (DSS) approach to the project scheduling problem is most appropriate. For a review of DSS problems see (Anthonisse et al., 1988), (Sprague, 1987) and (Sprague and Carlson, 1982). It is beyond the scope of this paper the discussion of the design of the DSS. Instead we will focus on some aspects of its algorithmic core.

The paper is organized as follows. In section 2 the model is formalized. Section 3 is devoted to the decomposition approach we use for the solution of the global problem. One of the subproblems identified this way is an assignment problem presented in section 4. The related problem of time window selection is investigated in section 5. Finally sections 6 and 7 are devoted to the mathematical characterization of the assignment problem and to a solution method.

2 Description of the problem

As already mentioned, the problem we deal with is a resource constrained scheduling problem which can be seen as a generalization of the job shop problem. Let us describe it more in detail.

A set of projects $\mathbf{P}$ and a set of resources $\mathbf{R}$ are given, where a project P consists of a set of tasks $T(P)$. Let us indicate by $\mathbf{T} := \cup_{P\in\mathbf{P}} T(P)$ the set of all tasks. Tasks require resources for their execution. No preemption of tasks is allowed.

Each resource $R \in \mathbf{R}$ is a renewable machine-type resource and can execute at most one task at a time.

For each task T there exist $n(T)$ alternative ways of processing the task, called *alternatives*. Each alternative W is characterized by a resource set $\mathbf{R}(W) \subset \mathbf{R}$ and by a processing time $p(W)$. The set of alternatives associated with the task T is denoted by $\mathbf{W}(T)$ and $\mathbf{W}(\mathbf{S})$ denotes the set of alternatives associated with the tasks of $\mathbf{S}$, i.e. $\mathbf{W}(\mathbf{S}) := \cup_{T\in\mathbf{S}} \mathbf{W}(T)$. The set $\mathbf{T}$ includes also two dummy tasks T_s and T_t called the starting and ending tasks respectively, for which it is conventionally assumed that $n(T_s) = n(T_t) = 1$, $\mathbf{R}(\mathbf{W}(T_s)) = \mathbf{R}(\mathbf{W}(T_t)) = \mathbf{R}$ and $p(\mathbf{W}(T_s)) = p(\mathbf{W}(T_t)) = 0$.

In other words the processing of task T according to the alternative W requires $p(W)$ time units and all resources in $\mathbf{R}(W)$ are at disposal of the task during its processing. It is expected in a typical situation that the bigger $\mathbf{R}(W)$ is the faster the processing is, so what is gained in terms of processing time is lost in terms of resource availability for the other tasks.

A release date $a(T)$ and a deadline $b(T)$ are associated with each task T; these quantities identify the time interval in which the task should be processed.

As usual in project scheduling, precedence constraints can be defined between pairs of tasks of the same project. Let us indicate by $G = (\mathbf{T}, C)$ the resulting on-node activity graph, with $\mathbf{T}$ the set of nodes identified with the set of tasks. An arc $(T, S) \in C$ exists whenever it is required that task T precedes task S. The graph G is obviously assumed to be acyclic.

A partial order relation $(\prec)$ is induced by the graph G on the set $\mathbf{T}$: $T \prec S$ whenever a path exists in G from node T to node S.

Indicating by $\mathcal{R}$ the set of real numbers, a schedule is a pair of functions

$$(F,t): \mathbf{T} \to \mathbf{W} \times \mathcal{R}$$

(with F the resource assignment and t the time assignment) such that

$$\begin{array}{ll}
F(T) \in \mathbf{W}(T), & \text{(assignment constraints)} \\
a(T) \leq t(T) \leq b(T) - p(F(T)), & \text{(task constraints)} \\
t(T) + p(F(T)) \leq t(S) \quad \text{whenever } T \prec S, & \text{(precedence constraints)} \\
\text{either} \quad t(T) + p(F(T)) \leq t(S) & \text{whenever } \mathbf{R}(F(T)) \cap \mathbf{R}(F(S)) \neq \emptyset, \\
\text{or} \qquad t(S) + p(F(S)) \leq t(T). & \text{(disjunctive constraints)}
\end{array}$$

The objective is to find a schedule minimizing the makespan

$$t_M(F,t) := t(T_t) - t(T_s).$$

The dependence of the makespan on the schedule has been explicity stated.

When $n(T) = 1$ and $|\mathbf{R}(W)| = 1$, $\forall T \in \mathbf{T}, \forall W \in \mathbf{W}$, and all tasks in a project are linearly ordered, then the problem reduces to a job shop problem. As a generalization of the job shop problem, the problem previously defined, which will be referred to in the following as $\mathcal{P}$, is NP-complete and computationally very hard.

3 A decomposition approach

First note that problem $\mathcal{P}$ can be simply stated as

$$\min_{(F,t)} t_M(F,t). \tag{1}$$

But, due to the particular definition of F and t, (1) can be rewritten as

$$\min_F \min_t t_M(F,t). \tag{2}$$

In fact a schedule can be given by first specifying the resource assignment and then specifying the time assignment on the basis of the resources and the relative processing times. Note that no time assignment can be specified independently of the resource assignment.

Denoting by $\mathcal{P}(F)$ the following problem

$$t_M(F) := \min_t t_M(F,t)$$

problem $\mathcal{P}$ can be succinctly written as

$$t_M := \min_F t_M(F) \tag{3}$$

Problem $\mathcal{P}(F)$ can be described as follows. Given a disjunctive graph $G = (N, C \cup D)$, where D is the set of disjunctive arcs such that $(T, S) \in D$ if and only if T and S have been assigned conflicting alternatives, i.e. $\mathbf{R}(F(T)) \cap \mathbf{R}(F(S)) \neq \emptyset$, find a time assignment of the tasks which minimizes the makespan. In other words, the problem has the characteristics of a job shop problem with a general structure of the graph G.

In this paper we are not concerned with problem $\mathcal{P}(F)$ and so we assume that a heuristic algorithm solving $\mathcal{P}(F)$ is available. In fact, such an algorithm exists for the job shop case (see, for instance Adams et al., 1988, Serafini et al., 1989) and can be extended to the more general case covered by problem $\mathcal{P}(F)$.

The formulation (3) suggests that a solution of problem $\mathcal{P}$ can be found by enumerating the functions F and by solving the corresponding problems $\mathcal{P}(F)$. However, the computational complexity of the problem makes in general the exhaustive enumeration practically impossible.

So our approach calls for solving $\mathcal{P}(F)$ only for a small subset of possible resource assignments. This subset will be found through an iterative process in which alternatively an assignment F is specified and a time assignment is found through $\mathcal{P}(F)$. The key idea for the iteration is the one of choosing F taking advantage of the information provided by the solution of the previous problems $\mathcal{P}(F)$.

So we are faced with the following problems: how to recover useful information from a schedule, how to use this information to find a new resource assignment, and when to stop the iterative process. In the next sections we will consider separately these problems.

4 The assignment problem

In this section we will discuss the problem of how to find a new resource assignment. In particular we will define a subset of tasks $\mathbf{S} \subset \mathbf{T}$ and then we will find in a suitable way a restricted assignment function $F(T)$ for all $T \in \mathbf{S}$. The problem of finding such an assignment will be denoted by $\mathcal{A}(\mathbf{S})$.

The reason why we want to assign the resources only to a particular subset of tasks will be clear in the next section. The case $\mathbf{S} = \mathbf{T}$ corresponds to the first iteration, when no information from a schedule is available.

Let us first consider the case $\mathbf{S} = \mathbf{T}$.

The approach to the assignment problem we present is simply based on the observation that the optimal makespan is greater than the busy time of each resource, that is:

$$t_M \geq \max_{R \in \mathbf{R}} p_R \tag{4}$$

where p_R is the total amount of processing time during which the resource R is actually used.

In the case in which there are no precedence constraints among the tasks and each task uses one resource:

$$t_M = \max_{R \in \mathbf{R}} p_R. \tag{5}$$

Whenever (5) holds, the problem of minimizing the makespan is equivalent to the problem of minimizing $\max_{R \in \mathbf{R}} p_R$. Of course, in the general case this is not true, so that the following model, in which the maximum resource use is minimized, only provides a heuristic assignment to problem $\mathcal{P}$.

So we may define the following

Problem $\mathcal{A}(\mathbf{T})$:

$$\min_F \max_{R \in \mathbf{R}} \sum_{T:\, R \in F(T)} p(F(T))$$

Problem $\mathcal{A}(\mathbf{T})$ can be also stated as a 0-1 linear programming problem. Let us define a $|\mathbf{R}| \times |\mathbf{W}|$ matrix A, in the following way:

$$a_{RW} := \begin{cases} p(W) & \text{if } R \in \mathbf{R}(W) \\ 0 & \text{otherwise} \end{cases}$$

and the assignment variables x_W as follows:

$$x_W := \begin{cases} 1 & \text{if } F(T) = W \\ 0 & \text{otherwise.} \end{cases}$$

Thus the assignment model can be stated as follows:

$$\begin{aligned} y_M = \ & \min \ y \\ & \sum_{W \in \mathbf{W}} a_{RW} x_W \le y \quad \forall R \\ & \sum_{W \in \mathbf{W}(T)} x_W = 1 \quad \forall T \\ & x_W \in \{0,1\} \quad \forall W \end{aligned}$$

The solution from Problem $\mathcal{A}(\mathbf{T})$ can then be used to solve problem $\mathcal{P}(F)$. If it turns out that the optimal makespan $t_M(F)$ given by problem $\mathcal{P}(F)$ is equal to y_M, i.e. equation (5) holds, then the schedule (F, t) found this way is certainly optimal.

However, this should be regarded as an exceptional case. In general the presence of conjunctive and disjunctive arcs makes $t_M(F) > y_M$. As a result, there can be time intervals in which the resource distribution is unsatisfactory.

In order to understand the problem let us first note that the problem $\mathcal{A}(\mathbf{T})$ tries to settle the competition of tasks for the same resources all over the time horizon of the process. Actually this is in some case unnecessary because tasks whose starting times are far apart in time do not really compete even if they share some resources. Intuitively two tasks do compete for a common resource only if they are so close in time that their processing times tend to overlap, but cannot do it because of the common resource and so they are forced to a disjunctive precedence constraint.

Furthermore due to the presence of release dates and deadlines, certain classes of tasks are always separated in time.

These considerations lead us to focus our attention to specially selected time intervals, called *time windows* in the sequel, and therefore to solve again an assignment problem, which is in this case restricted to a subset S of tasks whose processing times, as given by problem $\mathcal{P}(F)$ overlap with the time window.

5 Time windows

In this section we investigate the problem of defining a suitable time window on the basis of the current schedule. In principle our policy is to leave the choice of the time window to the decision maker, although the possibility exists of making such a choice automatic.

The choice has to be based on some criteria, and the following paragraphs will be devoted to the discussion of these criteria.

As already mentioned, strict inequality holds in (4) because of the presence of precedence arcs, which do not allow parallel processing of the involved tasks, whilst this is implicitly allowed in the formulation of problem $\mathcal{A}(\mathbf{T})$.

Let us note the difference between conjunctive and disjunctive arcs. The conjunctive arcs constitute intrinsic constraints of the problem and therefore the task sequencing induced by them is fixed independently of the schedule. A conjunctive arc defines a separation in time between two tasks and their predecessor and successor classes respectively, in such a way that there is no conflict in assigning the same resource to tasks in different classes. This suggests reassignments of resources restricted to either class of tasks.

The disjunctive arcs can be removed by a different resource assignment, which avoids conflicts, thus providing a possibly tighter schedule.

In both cases (conjunctive and disjunctive arcs) the reassignment of resources cannot be done without taking into account all tasks processed during the same period of time; otherwise the improvement for some tasks could be obtained at the expense of other tasks.

Among the many conjunctive and disjunctive arcs of the graph, we think that it may be sufficient consider only those lying on the critical path.

Let us discuss in detail a number of criteria for the selection of time windows. Once a time window has been identified by means of one of the presented criteria, a new assignment problem $\mathcal{A}(\mathbf{S})$ will be solved restricted to the set of tasks $\mathbf{S}$ which are scheduled in a time interval that overlaps with the time window.

1) Take a conjunctive arc on the critical path and consider as time window the time interval during which one of the two tasks involved is scheduled.

The previous discussion motivates this selection rule. Its soundness should be made evident by the following example.

A set of tasks $\mathbf{T}$ is given which consists of two tasks T and S, that is $\mathbf{T} = \{T, S\}$. A precedence constraint between the tasks is defined so that $T \prec S$. The resource set is $\mathbf{R} = \{R_1, R_2\}$. The alternatives of task T are $\mathbf{W}(T) = \{W_1^T, W_2^T\}$. For what concerns the first alternative of task T, $\mathbf{R}(W_1^T) = \{R_1\}$ and $p(W_1^T) = 10$, while the second alternative of task T is such that $\mathbf{R}(W_2^T) = \{R_1, R_2\}$ and $p(W_2^T) = 6$. For what concerns task S, $\mathbf{W}(S) = \{W_1^S, W_2^S\}$, with $\mathbf{R}(W_1^S) = \{R_2\}$ and $p(W_1^S) = 10$, $\mathbf{R}(W_2^S) = \{R_1, R_2\}$ and $p(W_2^S) = 6$.

In this simple case, the optimal schedule can be immediately obtained by $F(T) = W_2^T$ and $F(S) = W_2^S$. In the following we will show how the optimal solution can be found through a decomposition algorithm which makes use of rule 1) for the time window identification.

First of all, the solution of Problem $\mathcal{A}(\mathbf{T})$ gives the assignment $F(T) = W_1^T$ and $F(S) = W_1^S$ with $y_M = 10$. Obviously, due to the presence of the conjunctive arc, the solution of $\mathcal{P}(F)$ provides a schedule with $t_M(F) = 20$. On the basis of rule 1) the time window [0,10] is selected and problem $\mathcal{A}(\{T\})$ is solved providing $F(T) = W_2^T$. The new schedule is such that $t_M(F) = 16$. A subsequent application of rule 1) identifies the time window [6,16], and the solution of $\mathcal{A}(\{S\})$ provides $F(S) = W_2^S$ and the new schedule, with $t_M(F) = 12$, is optimal.

2) Take a chain of disjunctive arcs on the critical path, that link tasks $T^1, \ldots, T^k$ such that

$$\bigcap_{i=1,\ldots,k} \mathbf{R}(F(T_i)) \neq \emptyset.$$

The time window is identified as the time interval during which the tasks on the chain are scheduled, that is $[t(T_1), t(T_k) + p(F(T_k))]$.

As the previous discussion pointed out, the presence of a disjunctive arc on the critical path shows that at least two tasks are actually competing for a resource in that time interval. In general a chain of such arcs reveals a critical time period for a resource, so that a better resource distribution in the time window identified through criterion 2) can improve the resulting schedule. The following example should clarify the concept.

Let us consider a set of tasks $\mathbf{T} = \{T, S, U\}$ among which no precedence constraint is settled. The resource set is $\mathbf{R} = \{R_1, R_2, R_3\}$. A single alternative is given for tasks T and S, that is $\mathbf{W}(T) = \{W_1^T\}$ and $\mathbf{W}(S) = \{W_1^S\}$ with $\mathbf{R}(W_1^T) = \{R_1, R_2\}$, $\mathbf{R}(W_1^S) = \{R_1, R_3\}$ and $p(W_1^T) = p(W_1^S) = 10$. As far as task U is concerned, two alternatives are given, that is $W(U) = \{W_1^U, W_2^U\}$. While alternative W_1^U requires two resources, so that $\mathbf{R}(W_1^U) = \{R_2, R_3\}$ with $p(W_1^U) = 10$, alternative W_2^U requires a single resource and is somewhat slower, that is $\mathbf{R}(W_2^U) = \{R_2\}$ with $p(W_2^U) = 15$.

Problem $\mathcal{A}(\mathbf{T})$ prefers for task U the faster alternative W_1^U obtaining $y_M = 20$. Unfortunately the makespan obtained through problem $\mathcal{P}(F)$ is $t_M(F) = 30$, as the tasks have to be sequenced because of the structure of the resource sets. As the time window [0,20] identifies the tasks T and S for which a single alternative is given, let us consider the time window (10,30] selected through application of rule 2). Solution of a new assignment problem provides $F(U) = W_2^U$ with $y_M = 15$. Then the optimal schedule is obtained with $t_M = 25$.

The latter example shows a case in which the assignment problem $\mathcal{A}(\mathbf{T})$ fails to find the optimal solution because of the resource set structure and rule 2) can be successfully applied. We will present now an example in which the failure is due to the presence of a conjunctive arc and rule 2) can be still successfully applied.

Let us consider a set of tasks $\mathbf{T} = \{T, S, U\}$ with the following precedence constraints: $T \prec S$ and $T \prec U$. The resource set is $\mathbf{R} = \{R_1, R_2\}$. For what concerns task T, $\mathbf{W}(T) = \{W_1^T\}$ with $\mathbf{R}(W_1^T) = \{R_1\}$ and $p(W_1^T) = 20$. Two alternatives are given for both tasks S and U, $\mathbf{W}(S) = \{W_1^S, W_2^S\}$ and $\mathbf{W}(U) = \{W_1^U, W_2^U\}$. The first alternative for both tasks is such that $\mathbf{R}(W_1^S) = \mathbf{R}(W_1^U) = \{R_2\}$ with $p(W_1^S) = p(W_1^U) = 10$, while the second alternative is such that $\mathbf{R}(W_2^S) = \mathbf{R}(W_2^U) = \{R_1\}$ with $p(W_2^S) = p(W_2^U) = 1$.

Problem $\mathcal{A}(\mathbf{T})$ distributes resources as uniformly as possible among all tasks providing $F(T) = W_1^T$, $F(S) = W_1^S$ and $F(U) = W_1^U$. Solution of $\mathcal{P}(F)$ introduces an arc between tasks S and U so that $t_M(F) = 40$ which is not optimal. Rule 2) allows the identification of the time window (20,40] so that solution of problem $\mathcal{A}(\{S,U\})$ provides the optimal assignment with $F(S) = W_2^S$ and $F(U) = W_2^U$. The optimal schedule has makespan $t_M(F) = 21$.

Furthermore another criterion to be considered in the selection of time windows consists in taking into account release dates and deadlines, as already anticipated. There are many possible ways to exploit this information for the selection of time windows. One possibility consists in selecting as time window any slot given by two successive dates, no matter whether release dates or deadlines.

As already mentioned, the decision about the time window selection can be left to the decision maker, and it is therefore important to provide the decision maker with some tools in order to enhance his decisions. We limit ourselves to suggest two possible tools. One consists in displaying the critical path in a graphical form by using typical GANTT charts and another one in displaying the number of used resources as a function of time (or alternatively the patching one over the other of GANTT charts relative to the resources). The hollow zones of this function represent underutilization of the resources and therefore could be used for the identification of a time window.

6 A characterization of problem $\mathcal{A}(\mathbf{S})$

In this section we characterize the assignment problem $\mathcal{A}(\mathbf{S})$ in order to design an algorithm for its solution. First note that the particular case in which each task can be processed by any (singleton) resource with the same processing time for all resources (so that $|\mathbf{R}(W)| = 1$, $\forall W \in \mathbf{W}(T)$, $\cup_{W\in\mathbf{W}(T)}\mathbf{R}(W) = \mathbf{R}$ and $p(W) = p_T$, $\forall W \in \mathbf{W}(T)$) is the *Bin Packing Problem*, with each resource interpreted as a 'bin' and p_T the size of the 'item' T to be inserted into some bin (see Garey and Johnson, 1979).

Therefore Problem $\mathcal{A}(\mathbf{S})$ is NP-complete and there is no hope for an efficient algorithm. We shall develop a branch-and-bound type algorithm with LP relaxation.

So let us consider the following Problem $\overline{\mathcal{A}(\mathbf{S})}$

$$
\begin{aligned}
\overline{y}_M = \ & \min y \\
& \sum_{W\in\mathbf{W}(\mathbf{S})} a_{RW}x_W \le y && \forall R \in \mathbf{R} \\
& \sum_{W\in\mathbf{W}(T)} x_W = 1 && \forall T \in \mathbf{S} \\
& x_W \ge 0 && \forall W \in \mathbf{W}(\mathbf{S})
\end{aligned}
$$

The optimal solution of $\overline{\mathcal{A}(\mathbf{S})}$ is not integral in general, so its assignment variables x_W do not define an assignment for those tasks T for which there exists $W \in \mathbf{W}(T)$ such that $0 < x_W < 1$, i.e. x_W is fractional. In the following we investigate the number of integral solutions of $\overline{\mathcal{A}(\mathbf{S})}$.

First note that the above linear programming problem, once converted into standard form, has $|\mathbf{W}(\mathbf{S})| + |\mathbf{R}| + 1$ variables (including slacks) and $|\mathbf{R}| + |\mathbf{S}|$ rows. Let k be the number of active constraints in the optimal solution of $\overline{\mathcal{A}(\mathbf{S})}$. Note that the variable y, being unconstrained, must be in any basis, so that $|\mathbf{R}| + |\mathbf{S}| - 1$ variables, among the assignment and the slack ones, must be in any basis.

The slack variables corresponding to non active constraints must be in the optimal basis, so at most

$$|\mathbf{R}| + |\mathbf{S}| - 1 - (|\mathbf{R}| - k) = |\mathbf{S}| + k - 1$$

assignment variables are in the optimal basis.

Note also that the assignment constraints imply that at least one assignment variable per task must be strictly positive and thus in the optimal basis. It follows that for at most $(k-1)$ tasks there are fractional assignment variables and consequently the optimal solution of $\overline{\mathcal{A}(\mathbf{S})}$ is an assignment for at least $(|\mathbf{S}| - k + 1)$ tasks. Hence at most

$$|\mathbf{S}| + k - 1 - (|\mathbf{S}| - k + 1) = 2k - 2$$

optimal assignment solutions of $\overline{\mathcal{A}(\mathbf{S})}$ are fractional.

It is interesting to consider the dual problem of $\overline{\mathcal{A}(\mathbf{S})}$:

$$\begin{aligned}
\bar{y}_M = \ & \max \sum_{T \in \mathbf{S}} v(T) \\
& v(T) \le p(W(T)) \cdot \sum_{R \in W(T)} u(R) \qquad \forall W(T), \ \forall T \\
& \sum_{R \in \mathbf{R}} u(R) = 1 \\
& u(R) \ge 0
\end{aligned}$$

Denoting $\bar{u}(W) := \sum_{R \in W} \bar{u}(R)$, with $\bar{u}(R)$ the optimal dual relative to the resource constraints, the optimal linear programming value can be written as

$$\bar{y}_M = \sum_{T \in \mathbf{S}} \min_{W \in \mathbf{W}(T)} p(W)\bar{u}(W) \tag{6}$$

Equation (6) has a direct interpretation: each variable $\bar{u}(R)$ measures the scarcity of the resource R with respect to the other resources. Note that $0 \le \bar{u}(R) \le 1$, so this measure is expressed as percentage. A value $\bar{u}(R) = 0$ means that the use of resource R does not effect the optimal value and so the scarcity of R is null. On the contrary $\bar{u}(R) = 1$ means that the resource R is fully responsible for the optimal value (note that in this case all other optimal duals $\bar{u}$ are equal to zero because of the constraint $\sum \bar{u}(R) = 1$).

By summing $\bar{u}(R)$ over the resources actually employed by the alternative W we get the quantity $\bar{u}(W)$ which may be called the *utilization factor* of the alternative W. It is as if all resources had been subsumed by one single fictitious resource and alternative W used this resource at a level given by the utilization factor $\bar{u}(W)$. The processing time of the fictitious resource applied to alternative W is given by the product of the 'true' processing time $p(W)$ times the utilization factor $\bar{u}(W)$.

Then, among the alternatives $W \in \mathbf{W}(T)$ one has to choose the one minimizing the fictitious processing time $\bar{u}(W)p(W)$. If this minimum is unique then the corresponding primal variables are integral and give rise to an assignment.

7 An algorithm for the assignment problem $\mathcal{A}(\mathbf{S})$

As already anticipated the algorithm we develop is of a branch-and-bound type. The search tree has levels corresponding to the tasks and the branches outgoing from a node correspond to the alternatives of a certain task. Therefore each node of the search tree is characterized by the fact that some tasks have forced assignments. The successors of a node inherite the same forced assignments and a task, not yet assigned, receives forced assignments to different alternatives for all successors. The root of the search tree has no forced assignments.

Denoting by $\mathbf{T}_0$ the set of forcedly assigned tasks and by $\mathbf{T}_1$ the other tasks in a certain node s of the search tree, and by $F_s(\mathbf{T}_0)$ the forced assignments corresponding to node s, the following linear programming problem has to be solved on s:

$$\begin{aligned}
\overline{y}_M(s) = \quad & \min\ y \\
& \sum_{W\in\mathbf{W}(\mathbf{T}_1)} a_{RW}x_W \leq y - \sum_{W\in F_s(\mathbf{T}_0)} a_{RW} && \forall R \in \mathbf{R} \\
& \sum_{W\in\mathbf{W}(T)} x_W = 1 && \forall T \in \mathbf{T}_1 \\
& x_W \geq 0 && \forall W \in \mathbf{W}(\mathbf{T}_1)
\end{aligned}$$

The first computational experience shows that it is convenient to branch over that task whose solution is most fractional, i.e. $\max_{W\in\mathbf{W}(T)} x_W$ is minimal.

In order to speed up the computation it is also convenient to keep fixed the values of those variables that are integral in the previously solved lp problems. According to the considerations of the previous section we are left with a smaller system of at most $2|\mathbf{R}|-2$ assignment variables and $|\mathbf{R}|-1$ tasks. Although this does not guarantee optimality of the final solution, it definitely speeds up the computation.

8 Conclusions

In this paper we adopted for the solution of the resource constrained scheduling problem a decomposed approach that separates the assignment from the scheduling problem and foresees the interaction between algorithms and human experience. The motivations for such an approach are the complexity of the problem and the presence in scheduling problems of aspects hard to be formalized. It is believed that the more complex the problem the more advantageous this approach is.

Future research will be devoted to the testing of the presented approach, both from the algorithmic and the DSS design point of view. Moreover, the approach will be extended to more general problems, that for instance explicitly take into account different optimization criteria.

References

Adams, J., Balas, E. and Zawack, D. (1988). The Shifting Bottleneck Procedure for Job

Shop Scheduling. *Management Science*, 34, pp. 391–401.

Anthonisse, J. M., Lenstra, J. K. and Savelsbergh, M. W. P. (1988). Behind the Screen: DSS from an OR Point of View. Report OS-R8805, Centrum voor Wiskunde en Informatica at Amsterdam.

Bellman, R., Esogbue, A. O. and Nabeshima, I. (1982). Mathematical Aspects of Scheduling & Applications. Pergamon Press.

Conway, R. W., Maxwell, W. L. and Miller, L. W. (1967). Theory of Scheduling. Addison-Wesley, Reading, MA.

Davis, E. W. (1973). Project Scheduling under Resource Constraints: Historical Review and Categorization of Procedures. *AIIE Transactions* 5(4), pp. 297–313.

Davis, E. W. and Patterson, J. H. (1975). A Comparison of Heuristic and Optimum Solutions in Resource-Constrained Project Scheduling. *Management Science* 21(8), pp. 944–955.

French, S. (1982). Sequencing and Scheduling: an Introduction to the Mathematics of the Job-Shop. Horwood, Chichester.

Garey, M. R. and Johnson, D. S. (1979). Computers and Intractability. A Guide to the Theory of NP-Completeness. Freeman, San Francisco.

Lenstra J. K. and Rinnooy Kan, A. H. G. (1985). Sequencing and Scheduling. In: O'hEigeartaigh, M., Lenstra J. K. e Rinnooy Kan, A. H. G. (eds.), Combinatorial Optimization: Annotated Bibliographies. John Wiley & Sons.

Lawler, E. L., Lenstra, J. K. and Rinnooy Kan, A. H. G. (1982). Recent Developments in Deterministic Sequencing and Scheduling. In: Dempster, M. A. H., Lenstra, J. K. e Rinnooy Kan, A. H. K. (eds.), Deterministic and Stochastic Scheduling. Reidel, Dordrecht.

Olaguibel R. A. and Goerlich, J. M. T. (1988). Computational Comparison of Classical and New Heuristic Algorithms for Resource-Constrained Project Scheduling. Manuscript.

Patterson, J. H. (1984). A Comparison of Exact Approaches for Solving the Multiple Constrained Resource, Project Scheduling Problem. *Management Science* 30(7), pp. 854–867.

Rinnooy Kan, A. H. K. (1976). Machine Scheduling Problems: Classification, Complexity and Computations. Nijhoff, The Hague.

Serafini, P., Ukovich, W., Kirchner, H., Giardina, F. and Tiozzo F. (1989). Job Shop scheduling: a case study. In: Archetti, F., Lucertini, M. and Serafini, P. (eds.), Operations Research Models In Flexible Manufacturing Systems. Springer, Vienna.

Sprague, R. H., Jr. (1987). DSS in Context. *Decision Support Systems* 3(3), pp. 197–202.

Sprague, R. H. and Carlson, E. D. (1982). Building Effective Decision Support Systems. Prentice Hall, Englewood Cliffs.

Multicriterial Problems of Optimization of Industrial and Water-Protective Complexes Development

G.A. Sukhorukov
All–Union Research Institute of Water Protection
Kharkov, USSR

The concept of co-ordinated management of industrial development and nature conservation is gaining in importance as a component of modern ideas on harmonious development of civilization. The concept provides for interrelated development of industrial complexes and that for management and utilization of wastes, i.e. water-protective complexes. this would suggest that objectives of industrial complexes development (i.e. assigned needs in products and services) and objectives of nature conservation complexes development (e.g. assigned limits to discharge of harmful substances to the environment) can be attained either with minimum costs or with the best approximation when the development resources are limited.

Let us refer to a development program as $x(t)$, $t \in T$ vector, comprising the following vectors: $y(t)$ — outputs; $v(t)$ — operative controls; $z(t)$ — system's state; $u(t)$ — development controls. These variables have the following physical sense. If vectors $y(t)$, $v(t)$ for the industrial complex define, correspondingly, final and complete outputs (flows) of products, then vectors $z(t)$, $u(t)$ define capacities of the complex units and the units capacity increment. For environment protection complexes vector may define flows of wastes, and $y(t)$-flows of treated (rendered harmless) wastes discharged into environment. So, the operative control vector $v(t)$ define control under unaltered structure and parameters of the system, while $u(t)$ vector define a charge of the structure and parameters of the system.

Let us consider a general mode of development programs optimization models' synthesis, based on the principle of completion of the initial system. A model of the control program optimization with N criteria of optimality is considered as the initial system

$$W \subset T \times V \times Y, \qquad f_n^w : W \to \mathcal{F}, \qquad n = \overline{1,N}, \tag{1}$$

$$w_n^* \in \arg \min_{w \in W} f_n^w(w), \qquad f_n^w(w_n^*) \leq f_n^w(w), \qquad \forall w \in W, \tag{2}$$

where T — discrete aggregate of time moments; W — set of admissible programs of operative control; V — aggregate of operative controls; Y — set of outputs; f_n^w — scalar function (optimality criteria); $\mathcal{F} \subset \mathbf{R}^1$ — aggregate of quality programs evaluations; w_n^* — optimal vector relative to f_n^w critertia.

The synthesis of model development programs optimization is performed as follows. An intermediate system based on (1) is formed:

$$X_0 \subset T \times \{\emptyset\} \times \{z_0\} \times V \times Y, \qquad f_n : X_0 \to \mathcal{F}, \qquad n = \overline{1, N}. \tag{3}$$

Then is defined a so-called completed system

$$X \subset T \times U \times Z \times V \times Y, \qquad f_n : X \to \mathcal{F}, \qquad n = \overline{1, N}, \tag{4}$$

At that

$$x_{on}^* \in \arg \min_{x_0 \in X_0} f_n(x_0), \qquad x_n^* \in \arg \min_{x \in X} f_n(x), \qquad n = \overline{1, N}. \tag{5}$$

Here $\{z_0\}$ is a set containing z_0 vector, formed of elements of subsets of the model of development of operative control programs optimization parameters; X_0, X — sets of acceptable development programs in intermediate and completed model, correspondingly; U — control set; Z — set of state, $z_0 \in Z$.

It is obvious that system (3) is equivalent to system (1), including values of quality indices. As system (3) is structurally equivalent to system (4) it can serve as a basis for comparison of (1) and (4) by quality indices.

Definition. Let us call the completed system (4) relative to (1) as developing, i.e. the model of development programs optimization if

$$\forall n \in N : f_n(x_n^*) \le f_n(x_{on}^*), \qquad \exists k \in N : f_k(x_k^*) < f_k(x_{ok}^*). \tag{6}$$

Let us now consider a class of discrete dynamic models of development programs optimization with additive control of development. For simplification we accept availability of a single optimality criterion

$$F(x, t) = \sum_{t=1}^{T} f(y, v, z, u, t) \to \min \tag{7}$$

$$Ly(t) \underset{(\le)}{\ge} \omega(t), \qquad t = \overline{1, T}, \tag{8}$$

$$y(t) = Ay(t-1) + Bv(t), \qquad y(t=0) = y(0), \qquad t = \overline{1, T}, \tag{9}$$

$$Dv(t) \le z(t), \qquad t = \overline{1, T}, \tag{10}$$

$$z(t) = Pz(t-1) + Qu(t), \qquad z(t=0) = z(0), \qquad t = \overline{1, T}. \tag{11}$$

Here L, A, B, D, P, Q are nonlinear operators in a general case, and in a partial case they are a matrices of constraints, and $\omega(t)$ represents doals of development.

Signs $\ge$, $\le$ are determined by the physical sense of goals $\omega(t)$. If $\omega(t)$ is a vector of final product then $\ge$, but if it is a vector of admissible discharge of pollutants then $\le$. The essence of problem (7–11) consist in finding a program $x^*(t) = (y^*(t), v^*(t), z^*(t), u^*(t))$ ensuring attainment of $\omega(t)$ goals at minimal costs (7), i.e. by the effectively criterion. But in a case of limited resources for development it is necessary in a certain sense to examine a reverse problem in relation to (7–11). It is done by introduction of a resources restriction system, e.g. on capital expenditure in the form

$$Cu(t) \le \beta, \qquad t = \overline{1, T}, \tag{12}$$

where C designates an operator or matrix, $\beta(t)$ is a given value of limited costs.

Moreover the system of limitations (8) determining directional multitude is transformed into a system of the form

$$Ly(t) \underset{(-)}{+} \rho(t) \underset{(\leq)}{\geq} \omega(t), \qquad t = \overline{1,T}, \tag{13}$$

where $\rho(t)$ is the closing errors vector.

Instead of the economical criterion (7) the accuracy criterion is introduced in the form

$$\Phi(\rho,t) = \sum_{t=1}^{T} \varphi(\rho,t) \to \min \tag{14}$$

It determines a degree of $\omega(t)$ goals attainment.

So, the problem of development program optimization under limited resources includes the limitation system (13, 9–11, 12) and the accuracy criterion (14).

For a wide scope of problems, important in practical context, the system (7–11, 12–14) can be sufficiently simplified, and a base model of the development programs optimization can be formulated in the form

$$F(v,u) = f_v(v(T)) + \sum_{t=1}^{T} f_u(u(t)) \to \min \quad \text{or} \quad \Phi(\rho) = \sum_{t=1}^{T} \varphi(\rho(t)) \ \to \ \min \tag{15}$$

$$Bv(t) \underset{(\leq)}{\geq} \omega(t) \quad \text{or} \quad Bv(t) \underset{(-)}{+} \rho(t) \underset{(\leq)}{\geq} \omega(t), \qquad t = \overline{1,T}, \tag{16}$$

$$Dv(t) - \sum_{\tau=1}^{t} u(\tau) \leq z(0,t), \qquad t = \overline{1,T}, \tag{17}$$

$$Cu(t) \leq \beta(t), \qquad t = \overline{1,T}, \tag{18}$$

where $z(0,t)$ represents a prescribed state of the system at a moment t, under a known state $z(0)$.

On the basis of (15–18) it is possible to formulate two mutually reversible problems

$$\begin{array}{ll} \gamma = \varphi(\rho) \to \min & \varphi(\rho) \leq \gamma \\ \bar{B}v + \rho \geq e & \bar{B}v + \rho \geq e \\ Dv - u \leq z & Dv - u \leq z \\ f(v,t) \leq \alpha & \alpha = f(v,u) \to \min \end{array} \tag{20}$$

where e — unit vector, $\bar{B}$ — matrix formed by division of B matrix into elements of ω vector.

On the basis of (19–20) analysis the following can be proved.

Lemma 1. Let $\alpha \in [0, \alpha^{\max}]$ then solution of (19) — (v^*, u^*, ρ^*) problem will be also the solution of problem (20) if in it $\gamma = \varphi(\rho^*)$. Inversely, if in (20) $\gamma \geq 0$ then solution of (20) — (v^0, u^0, ρ^0) problem will be the solution of (19) if in it $\alpha = f(v^0, u^0)$.

Corollary. Solution of the couple (19, 20) problems belongs to the Pareto's set relative to $\varphi(\rho), f(v,u)$ criteria.

Let us examine an often met case when for a long period of time $[0,T]$ a liner problem must be solved for a minimum of discounted costs, and for an intermediate time interval the problem with a quadratic accuracy criterion. Let us assume that the whole period of time consists of two intervals, when the couple of problems is of the form:

$$F(v,u) = c'_u\tilde{u} + c'_v v(2) \to \min \tag{21}$$

$$Bv(2) \geq \omega(2) \tag{22}$$

$$-\tilde{u} + Dv(2) \leq z(0,2) \tag{23}$$

$$\Phi(\rho) = c'_\rho \rho(1) + \rho'(1)G\rho(1) \to \min \tag{24}$$

$$Bv(1) + \rho(1) \geq \omega(1) \tag{25}$$

$$-u(1) + Dv(1) \leq z(0,1) \tag{26}$$

$$u(1) + u(2) = \tilde{u}^*, \qquad c'_u u(1) \leq \beta(1) \tag{27}$$

Here c'_u, c'_v, c'_ρ are transpositioned vectors of coefficients; G is the matrix; $\tilde{u}^*$ — an optimal vectors in (22–23) problem (here it is assumed that $u(1) + u(2) = \tilde{u}$).

On the basis of (21–27) it is possible to prove the following.

Lemma 2. As a result of solution of the couple of problem (21–23) and (24–27) the finite goals $\omega(2)$ are attainable at minimal cost in the sense of (21), and intermediate goals $\omega(1)$ with the best accuracy in the sense of (24). Let us assume that the problem (21–23) contains $N+1$ criterion of the form

$$F(v,u) = \sum_{n=1}^{N} f_n(v_n,u_n) \to \min, \qquad f_n(v_n,u_n) \to \min, \qquad n = \overline{1,N}, \tag{28}$$

and the problem (24–27) contains $N+1$ criterion of the form

$$\Phi(\rho) = \sum_{n=1}^{N} \varphi_n(\rho_n) \to \min, \qquad \varphi_n(\rho_n) \to \min, \qquad n = \overline{1,N}. \tag{29}$$

Here $n = \overline{1,N}$ is a number of subsystem, designated a corresponding optimality coefficient. As subsystems can be designated, e.g. a facility, region, or state. Then the essence of the problem consists in search for a compromise optimal solution, matching interests of N local subsystems with that of the systems as a "whole", i.e. reflecting general interests. Let us consider a method of search for compromise optimal solution, based upon the idea of parametrization (Mikhalevich and Volkovich, 1982; Sukhorukov and Tzybul'nik, 1978).

Let us define an assembling of normal local criteria specifying relative costs of separate subsystems for implementation of a program. Using the known scalarization method (Mikhalevich and Volkovich, 1982), we get

$$\psi_n = (v_n,u_n) = \frac{f_n(v_n,u_n) - f_n^{\min}}{f_n^{\max} - f_n^{\min}}, \qquad n = \overline{1,N}, \tag{30}$$

where $f_n^{\min}, f_n^{\max}$ are, correspondingly, minimal and maximal values of costs.

On the range of values of $\psi_n(v_n, u_n)$ criteria we designate a point ψ_0 corresponding to minimal and equal values of relative costs (30). It means that ψ_0 corresponds to an idea of "just" distribution of relative costs among subsystems. Let us complement the systems (21–23) with constraints

$$\psi_n(v_n, u_n) \leq \mu, \qquad \mu \in [\psi_0, 1], \qquad n = \overline{1, N}, \tag{31}$$

where μ — variable parameter. At $\mu = 1$ in the problem (21–23, 31) the best (minimal) value of summary cost (the global criterion) is attainable, and at $\mu = \psi_0$ we get the worst (maximal) value.

Introduction of some additional conditions will to obtain the only solution corresponding, e.g. to a compromise between a degree of global criterion deterioration and unevenness of relative costs by subsystems.

Similar procedure of scalarization and search for the compromise optimal solution is applicable to the system of criteria (29).

The results described above have provided a frame for development of a multistep procedure. The first step consist in elaboration of a program of industrial development and water protection, providing for attainment of the final goals (e.g. production target, water quality standards) at compromise optimal distribution of costs. At the second and following steps should be determined stages of the program implementation, e.g. measures providing for the best approximation to the end goals in the sense of correctness criterion. The practical result of implementation of the described procedure consists in development of problem-oriented packs of applied programs and their use for design of water-protective programs in basins of rivers: the Severski Donets, the Dnieper, the Oka (USSR), and the Connecticut (USA).

References

Mikhalevich, V.S. and Volkovich, V.L. (1982). Computing methods on research and design of complex systems. Nauka, 286p., (in Russian).

Sukhorukov, G.A. and Tzybul'nik, S.A. (1978). Prognostication and optimization of intensive water-protective measures by several criteria of optimality. Sbornik nauch. trudov, VNIIVO, Kharkov, vyp. 8, pp. 137–145, (in Russian).

Structural Design of Engineering Systems with Regard to Many Criteria

A.F. Voloshin, S.V. Volkovich, V.A. Zaslavsky
Kiev State University
Kiev, USSR

A two-staged procedure of optimal design of engineering system structure stated as follows. On the first stage on the set of elements of technical realization of subsystems of a designed system a variant of system is chosen with regard to qualitative solution of functional problems posed before it which are defined by the given set of subsystems. The necessity of optimal use of resources is also considered. On the second design stage a spatial arrangement of obtained on the first stage a variant of system in bounded volumes is made with regard for the criteria which define the length of a network connecting the subsystems arranged and the balance of the center of gravity of spatially distributed structure of a system.

The system consisting of n subsystems is examined. Each subsystem, j, $j \in J = \{1, 2, \ldots, n\}$ independently of others can be in one of two states: efficiency ($y_j = 1$) and refusal ($y_j = 0$). At the arbitrary time $t \in T$ system in the whole can be in one of the 2^n states $y = (y_1, y_2, \ldots, y_n)$ from the set of states Y each of those is characterized by the index of conditional probability $\phi(y)$ $(0 \leq \phi(y) \leq 1)$ of its functioning. The system is intended for carrying out the final number of problems $z \in Z = \{1, 2, \ldots, z_0\}$. Let $J_z = \{j_1, \ldots, j_z\}$, $j_1 \in J, \ldots, j_z \in J$ be a set of subsystems which are employed in the carring out the problem z. The subsystem j is realized on one of the $k \in K_j = \{1, 2, \ldots, k_j^*\}$ types of elements e_{jk} which are reserved with a multiplicity of reservation $\lambda_{jk} \in [\alpha_{jk}, \beta_{jk}]$, $\alpha_{jk}, \beta_{jk}, \lambda_{jk} \geq 0$ — integers. Then a set of possible variants of realization of j-th subsystem is determined as

$$V_j = \{\, v_j = (\lambda_{jk_1} e_{jk_1}, \ldots, \lambda_{jk_s} e_{jk_s}),\ \lambda_{jk_r} \in [\alpha_{jk_r}, \beta_{jk_r}],\ r = \overline{1, s},\ k \in K_j \,\}.$$

The elements e_{jk} have a reliability (probability of failproof performance on the interval of time T) $p_j(e_{jk})$ and value of resources exponents $g_{ij}(e_{jk})$, $i \in I$, $j \in J$, where $I = \{1, 2, \ldots, m\}$ — set of exponents of limited resources. The reliability of the variant v_j of j-th subsystem is defined depending on the way of reservation and types of elements in the subsystem (Volkovich et al., 1984; Volkovich et al., 1985; Volkovich and Zaslavsky, 1986). The value of resources exponents $g_{ij}(v_j)$ on the variant v_j are calculated starting from multiplicity of reservation and a number of types of elements as follows:

$$g_{ij}(v_j) = \sum_{r=1}^{s} (\lambda_{jk_r} + 1) \cdot g_{ij}(e_{jk_r}).$$

The problem of optimal design in multicriterial statement is formulated as follows:

$$P(p_1(v_1),\dots,p_n(v_n)) = \sum_{y\in Y}\phi(y)\cdot\prod_{j\in J}p_j(v_j)^{y_j}(1-p_j(v_j))^{1-y_j}\to\max \tag{1}$$

$$g_i(v) = \sum_{j\in J}g_{ij}(v_j)\to\min,\quad i\in I_0, \tag{2}$$

$$P_z(v^z) = \prod_{j\in J_z}p_j(v_j)\ge P_z^*,\quad z\in Z, \tag{3}$$

$$g_i(v) = \sum_{j\in J}g_{ij}(v_j)\le b_i,\quad i\in I, \tag{4}$$

$$v=(v_1,\dots,v_n)\in V = \prod_{j\in J}V_j,\quad v^z=(v_j)_{j\in J_z}. \tag{5}$$

Here P is the reliability of a system, $g_i(v),\ i\in I_0$ are exponents of optimized and $g_i(v)$, $i\in I$ limited resources, $I_0=\{1,2,\dots,l^*\}$ — set of optimized criteria, $P_z^*,\ z\in Z$ — assigned limitations on the probability of carrying out problems z.

The problem (1)–(5) may be considered also without limitations. In particular, as a criterion of reliability (1) the reliability function of a successive system may be given $P(v)=\prod_{j\in J}p_j(v_j)$ i.e. such a system in which a refusal of any subsystem results in refusal of a subsystem as a whole.

The vector of preference $\rho=(\rho_i)_{i\in I_0\cup\{0\}},\ \rho_i>0,\ \sum_{i\in I_0}\rho_i=1$ is given for the problems (1)–(5). This vector defines the importance of optimized criteria (Volkovich et al., 1984). Then the compromise solution is understood as a solution of multicriterial problems on which minimal and equallyweighted (with the help of preference coefficients) discrepancies with respect to all criteria given on a single scale (Volkovich et al., 1984) are attained.

Let $v^*=(v_1^*,\dots,v_n^*)$ be a compromise solution of the problem (1)–(5) with exponents $P(v^*)$, $P_z(v^{z^*})$, $g_i(v^*)$, $i\in I_0\cup I_1$ and technical-economical characteristics of subsystems variants $p_j(v_j^*)$, $g_{ij}(v_j^*)$, $i\in I_0\cup I_1$, $j\in J$.

The problem of spatial arrangement of compromise variant of a system structure $v^*=(v_1^*,\dots,v_n^*)$ obtained in limited sizes with minimized criteria of center of gravity and the length of a network connecting the arranged subsystems is solved on the second design stage.

Let S_r be technical cuttings, $r\in R=\{1,2,\dots,r^*\}$ with a given limitations by technico-economical parameters b_r^i, $i\in I_r$, $r\in R$ in size, for example. The matrix of distances $D=\|d_{r_1r_2}\|_{r_1r_2\in R}$ is given for the pairs of cuttings S_{r_1},S_{r_2}. Let $r(l)=\{j_1,j_2,\dots,j_l\}$ be a fixed combination of n elements by l elements which consists of subsystem indices $j_1,j_2,\dots,j_l$, $j_1<j_2<\dots<j_l$, $j_1,j_2,\dots,j_l\in L_{r_l}$ and
$A_{L_r}^l=\{\,r(l)\mid r(l)=\{j_1,\dots,j_l\},\ 1\le j_1<\dots<j_l\le n\,\}$, $l\in L_r$ — be a set of combination $r(l)$ by l indices $l=\overline{1,|L_r|}$. Here the set $L_r=\{1,2,\dots,n\}$ has the number of elements $|L_r|$. For each combination $r(l)$ the possible variant $U_{r(l)}=(v_{j_1}^*,v_{j_2}^*,\dots,v_{j_l}^*)$ of an arrangement of S_r cutting is determined. Then $U_r=\bigcup_{l\in L_r}\bigcup_{r(l)\in A_{L_r}^l}U_{r(l)}$ is a set of possible variants of an arrangement of S_r cutting. The limitations by resources for this cutting are given in the form

$$g_r^i(u_{r(l)}) = \sum_{\substack{\nu\in r(l)\\ r(l)\in A_{L_r}^l}} g_{i\nu}(v_\nu^*)\le b_r^i,\quad i\in I_r' \tag{6}$$

$$g_r^i(u_{r(l)}) = \sum_{\substack{\nu \in r(l) \\ r(l) \in A^l_{L_r}}} g_{i\nu}(v^*_\nu) \geq b^i_r, \quad i \in I''_r \tag{7}$$

where the functions of resources inputs $g_r^i(u_{r(l)})$ are determined by means of the combinations $r(l) \in A^l_{L_r}$, $I'_r \subset I$, $I''_r \subset I$, $r \in R$ — finite sets of limitations indices on S_r cutting. For a variant of a system the matrix of connectedness of subsystems $C = \|c_{\tau\nu}\|_{n \times n}$ is given in which elements are

$$c_{\tau\nu} = \begin{cases} 1, & \text{if subsystem } \tau \text{ is connected with the subsystem } \nu; \\ 0, & \text{if subsystem } \tau \text{ is not connected with the subsystem } \nu; \end{cases}$$

where $c_{\tau\nu} = c_{\nu\tau}$ and $c_{\tau\tau} = 0$, $\tau \in J$.

At the arrangement of subsystems it takes account of the condition of the subsystems' inability to joint which is assigned by the matrix $B = \|b_{\tau\nu}\|_{n \times n}$ with the entries

$$b_{\tau\nu} = \begin{cases} 1, & \text{if a subsystem } \tau \text{ joints with subsystem } \nu, \\ 0, & \text{if a subsystem } \tau \text{ doesn't joint with subsystem } \nu, \end{cases}$$

and also $b_{\tau\tau} = 0$.

In combinatorial statement, the problem of optimal arrangement with regard for above denotations is formalized as follows.

It is necessary to choose a variant of arrangement

$$u = (u_{1(l)}, u_{2(l)}, \ldots, u_{r(l)}) \in U = \prod_{r \in R} u_r, \tag{8}$$

of variant v^* in the cuttings S_r, $r \in R$, which minimizes the length of a network that connects subsystems

$$d(u) = \sum_{1 \leq i < j \leq r^*} d_{ij} \cdot \delta(u_{r_i(l)}, u_{r_j(l)}) \to \min, \tag{9}$$

where

$$\delta(u_{r_i(l)}, u_{r_j(l)}) = \begin{cases} 0, & \text{if } \sum\limits_{\substack{\tau \in r_i(l) \\ \nu \in r_j(l)}} c_{\tau\nu} = 0 \\ 1, & \text{if } \sum\limits_{\substack{\tau \in r_i(l) \\ \nu \in r_j(l)}} c_{\tau\nu} > 0 \end{cases}$$

and minimizes a deviation from the given center of gravity

$$f(u) = ((x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2)^{\frac{1}{2}} \to \min \tag{10}$$

where the coordinates x, y, z depend on the variant of arrangement u and are defined as follows:

$$\begin{aligned} x &= \frac{1}{g_{i_0}(v^*)} \cdot \sum_{r \in R} x_r \cdot g^r(u_{r(l)}), \\ y &= \frac{1}{g_{i_0}(v^*)} \cdot \sum_{r \in R} y_r \cdot g^r(u_{r(l)}), \\ z &= \frac{1}{g_{i_0}(v^*)} \cdot \sum_{r \in R} z_r \cdot g^r(u_{r(l)}), \end{aligned}$$

Here $g_{i_0}(v^*)$ is the value of mass of the system on a variant of structure v^*, where i_0 is the number of limitation or criterion. Note that the combinations $1(l), 2(l), \dots, r(l)$ do not intersect, i.e. they are built of the different indices of subsystems variants, but their join gives a whole variant of arrangement.

The examined above problems of structural design of engineering systems with regard to many criteria are solved in a dialogue system of automatized design (Volkovich et al., 1985).

The algorithms of solution of problems of multicriterial discrete mathematical programming in the statements (1)–(5) and (6)–(10) based on the method of successive analysis of variants. Procedures of analysis, elimination and construction of variants permitting to generate a compromise solution of the problem are taken as a principle of algorithms. The analysis procedures for the problems (1)–(5) come to the construction and investigation of join of succession of limitation systems of the form:

$$\begin{aligned}
P(p_1(v_1), \dots, p_n(v_n)) &\geq P^0 - \frac{k_0}{\rho_0}(P^0 - P_{(\min)}), && (11)\\
g_i(v) &\leq g_i^0 + \frac{k_0}{\rho_i}(g_{i(\max)} - g_i^0), \quad i \in I_0, && (12)\\
P_z(v^z) &\geq P_z^*, \quad z \in Z, && (13)\\
g_i(v) &\leq b_i, \quad i \in I, && (14)\\
v = (v_1, \dots, v_n) \in V &= \prod_{j \in J} V_j, \quad v^z = (v_j)_{j \in J_z}, && (15)
\end{aligned}$$

of purposefully selective values of the parameter $k_0 \in (0, 1)$. Here P_0, $g_{i(\max)}$ — are maximal, and $P_{\min}(v)$, $g_i^0(v)$ — minimal values of the functions $P(v)$, $g_i(v)$ correspondingly.

The procedures of elimination consist in exclusion of the consideration the variants of solutions which do not satisfy the system of limitations (11)–(13). This is done with the help of procedures of the componentwise elimination. The construction procedure is a choice on the narrowed sets of possible solutions a compromise variant with regard to generalized criterion

$$F(V) = \rho_0 \cdot P(v) + \sum_{i \in I_0} \rho_i g_i(v)$$

on the set

$$V' = \{\, v \mid \rho_0 \cdot \Delta P(v) \leq k_{0(\min)},\ \rho_i \Delta g_i(v) \leq k_{0(\min)},\ i \in I_0 \,\},$$

where $k_{0(\min)}$ is a minimal value of the parameter k_0, for which the system of inequalities (11), (12) is joint; $\Delta P(v) = P^0 - P(v)$, $\Delta g_i(v) = g_i(v) - g_i^0$, $i \in I_0$ — disagreements of criteria on the variant v.

In the problem of arrangement the limitations similar to (11), (12) are given for the criteria (9), (10) and a system of limitations (6)–(8) is considered with limitations on the criterion (9), (10).

The sets of possible arrangement variants construct at the expense of elimination of elements from the sets L_r which bring to reduction of the number of combinations $r(l)$ and reduction of the lengths of possible arrangement variants in the cuttings chosen. No explicit forms of sets are formed in the analysis of variants. If as a result of elimination a number of the arrangement variants remaining in the sets U_r are not large, then an explicit forming of the sets U_r is carried out, and the search of compromise solutions is accomplished on the remaining sets.

The algorithms the mathematical software is based on are dialogue and are realized on the high level languages.

The system SKIF is used as the system software for a dialogue system. It extends the means of computer's (Volkovich et al., 1985). It allows in the run of dialogue seance to synthesize from the base (Volkovich et al., 1984; Volkovich and Zaslavsky, 1986) the procedures of solution of the problems of the problem domain and organizes the system work in the whole.

A user who works with the system is given the following possibilities:

- to choice the types of models of a design object and its parameters. A designed object can be regarded as a successive or nonsuccessive system (Volkovich et al., 1984; Volkovich and Zaslavsky, 1986). Variants of subsystems can be built on the base of onetype or manytype elements, with reservation or without reservation and any of the criteria can be included in the number of optimized or act as a limitation;

- to reduce automatically the problem of optimization of a designed object structure to one of the canonical models of the problems of manycriterial discrete optimization and carry out its solution. The problems can be solved both automatically and in interactive condition;

- to deduce on the output the results of the problems solutions in the terms of design object;

- to form and modify in the iterative condition an information on the designed objects in the data files used in the system.

References

Volkovich, V.L., Voloshin, A.F., Gorlova, T.M. et al. (1984). Methods and algorithms of automated design of complex control systems. Kiev, Nauk. dumka, 216p.

Volkovich, V.L., Voloshin, A.F., Zaslavsky, V.A., Bondarchuk, Yu.V. (1985). System of automation design of complex system structures on the criterion of reliability /DISON/. *Control systems and machines*, p. 74–78.

Volkovich, V.L. and Zaslavsky, V.A. (1986). An algorithm of the solution of problems of optimization of a complex system reliability using in the subsystems manytype reserve elements. *Kybernetika*, 5, p. 54–61.

Part 3

Multiple Criteria Decision Support Systems

Decision Support Information System Development

V. Britkov
Institute for Systems Studies
USSR Academy of Sciences
Moscow, USSR

The complex problems investigated by computer science methods are as a rule multicriterial and poorly formalized.

To solve multicriterial problems the reduction to one criterion by reducing criteria with weight coefficient is necessary or the representation of a number of criteria as limitations is used. In this case it is difficult to formalize the choice of weight or limitations. Another way is to refuse from the attempts to reduce the problem to formal criteria and use human abilities to make decisions with many criteria in poorly formalized cases. When using this way a specialist (a decision-maker — DM) should receive the information represented in an optimum way. This optimum criterium sufficiently depends on the concrete DM. Decision Support System (DSS) solve this kind of problems.

The modern DSS is an integrated complex which must include the following elements: the Information System (IS), the Modelling System (MS), the EXPERT SYSTEM (ES), and intelligent interface tools which provide communication with final users (fig. 1).

IS includes the DATA BASE Management System (DBMS), Database (DB) and metadatabase, the data definition and manipulation languages, the intelligent interface, the information analysis and the like. The example of this IS is described in (Aleev, Britkov, Vasilyev and Shibaev, 1987).

MS includes the library of models, algorithms (the Model Base-MB) models program tools, calculation procedures generator, etc.

ES is system which allows to use non-formal methods of information development, to accumulate and use the knowledge of experienced specialist, to explain decisions applied and rules used, etc.

These 3 systems communicate, use specific methods relying on data-, knowledge- and model base. There are specific program tools and development methods for each of the systems described.

Let us examine these tools.

The developed DBMS, the communication and manipulation language are necessary for the information system development. The object-oriented approach with program systems Smalltalk-80, the tools of operation with multi-media database (Abul-Huda and Bell, 1988).

The functional programming is the most perspective in modelling systems development and use. The paper by Yurchenko (1987) is devoted to the modelling system development.

Presently the expert systems are the most popular. To be more precise not only ES, but a wider range, the artificial intelligence systems are used in DSS. To develop them there is a vast set of program tools such as the logical programming languages (PROLOG, superhigh level languages and Systems OPS-5, KEE, etc. (Savory, 1988).

The intelligent interface is the integral part of the artificial intelligence systems. It allows the end user to formulate tasks and inquires in a language similar to the a specialist in a concrete application domain. Special program tools — linguistic translator developed by means of the PROLOG language, are used to develop interfaces (Britkov and Aleev, 1988).

The DSS development methods

The development of DSS corresponding to the modern level of information technology development, is a complex and difficult task, which requires united effort of various specialists. Joining of their efforts must be based by means of so-called designing support systems, which must provide the computer support of the whole life cycle of DSS comprising projecting, realisation, use and accompanying. The example of these systems is CASE-technology (Chikofsky and Rubenstein, 1988) which is widely practiced in the present day.

Special methodologies and program means-generators DSS and instrumental systems are built for DSS development.

The development of the information system technology has recently caused the emergence of one more stage of the development — shells. The present technology is such that shells are developed by means of generators and then by means of special methods and appropriate program as well.

The following classification of the stages of information system development is presently adopted: the demonstration prototype, the research prototype, the experimental specimen, the industrial stage, the commercial product.

Usually the system gradually passes on from one stage to another, though a lot of them end up the first stage.We suggest another approach to solving a poorly formalized task which arises from the task of computerising the whole life cycle of the system development and use. The system consists of subsystems. The stage of the development of any subsystem may differ from the next. It depends on a number of circumstances: the degree of our knowledge and potentialities of the subtask decision, the degree of necessity and urgency (the computer speed limitations, the basic software non-perfection, resource limi-tations). Thus, some subsystems can reach the industrial stage and some can be at the prototype stage.

A number of tasks can pass on from the rank of poorly formalized tasks to the rank of formalized tasks just by increasing of the computer potentialities: speed and memory. The obvious case is chess: it is a poorly formalized task with limited computer speed which turns into a formalized one with the unlimited computer speed (in case it is possible to make exhausted search).

The quick development of the tools of problem specification, system development — logical, functional, object-oriented programming, the tendency to their integration makes the progress in the poorly formalized problem translation into the computer language possible.

The information system life cycle

The minimization of the time within the task emergence and the moment of the appropriate system introduction is necessary to effectively use the information systems. The possibility of the improvement of utilization methods and the appropriate modification of program products is also necessary. It is only possible by the system approach to the information means development and use (in particular, information systems) and the obligatory use of instrumental means of development support.

The structural scheme of the interaction of the information system and the design tools is represented in fig. 2.

According to the conventional methodology the first stage of the design is the subject area modelling. A subject area knowledgebase is developed at this which can be divided into three parts:

1. general knowledge
2. the specific subject matter knowledge
3. the research specific task knowledge.

The boundary between these databases are relative and are determined by the mode of use of proper knowledge. This knowledge is used at the following stages of development using formal and non-formal methods of designing (Cauvet et al. 1988). The E-R approach (Chen, 1976) has recently the most popular for the subject matter modelling of information systems. Despite of its limited nature and drawbacks it is widely used in projecting tasks. A vast score of attempts to develop and generalise the E-R approach are taken (Gutzwiller et al. 1988), but none of them has reached the stage of being well developed.

The next stage of the development is the analysis of demands to IS projected. A projected IS databases is formed at the stage of the analysis of demands. It includes functional demands which will use the IS data, for example the devices if information introduction and exit, data size, user categories, priorities and privileges, etc. The demands to user interface are formed. The latter includes the demands to the degree of user grounding, communication languages, etc.

The predecessors of the appropriate knowledge base are the systems of data rederence-dictionaries (Leong-Hong and Plagman, 1982).

A very important stage of the IS development is the analysis of the current of data and information development tasks. This stage is not considered in a due in the majority of existing methodologies (Information Systems Design Methodologies, 1982). The result of it is idea of the way the data are used, and what is more important — developed. The result of it is that the collection of the most data with emerging size problems, time of access, integrity, etc. become the task of top priority. At the same time the analysis of information development methods can reduce data sizes and escape a lot of problems dealing with the IS universal character.

To lounch conceptual, logical and physical design using the relational data model a set of systems (Information Systems Design Methodologies, 1982) is developed, the author taking part in them (Gelovani et al. 1986).

The subsequent stages of the IS life cycle depend considerably on the field of use and are not considered in the paper.

References

Abul-Huda, B. A. H. Bell, D. A. (1988). An Overview of a Distributed Multi-Media Database Management Systems (KALEID). In: Information Technology for Organisational Systems. Bullinger, H. J. et al. (eds.) Elsevier Science Publishers B. V. (North-Holland), pp. 443–449.

Aleev, V. Britkov, V. Vasilyev, A. Shibaev, S. (1987). Interactive Information Systems for Oceanological Data Processing. In.: Mathematical Methods of Informatics and Software in Systems Modelling. — Collections of papers. Vol. 15. Moscow: The Institute for Systems Studies, pp. 51–68 (in Russian).

Britkov, V. Aleev, V. (1988). An intelligent user interface for an environment information systems. In: Scandinavian Conference on Artificial Intelligence. Proceedings of the SCAI'88. IOS Amsterdam. Springfield, VA, pp. 33–41.

Cauvert, C. Proix, C. Rolland, C. (1988). Information systems design: an expert system approach. In.: The Role of Artificial Intelligence in Database and Information Systems. China Academic Press. Gungzhou, China, pp. 1–28

Chen, P. P. (1976). The Entity-Relalionship Model – Toward a Unified View of Data, *ACM Transactions on Database Systems*, No. 1, pp. 9–36.

Chikofsky, E., Rubenstein, B. L. (1988). CASE: reliability engineering for information systems. *IEEE Software*, vol. 5, No. 2, pp. 11–16.

Gelovani, V., Besrukov, D., Britkov, V., Golosov, A., Tsalenco, M. (1986). Interactive Design of Information Systems. The USSR Academy of Sciences Proceedings. *Technical Cybernetics*, No. 2, pp. 48–70 (in Russian).

Gutzwiller, Th. A., Lehmann-Kahler, M. G. C., Oesterle, H. (1988). Computer Based Strategic Information Planning Architecture. In: Information Technology for Organisational Systems. H. J. Bullinger et al. (Editors). Elsevier Science Publishers B. V. (North-Holland), pp. 347–353.

Information Systems Design Methodologies, (1983). A Feature Analysis, Proceedings of the IFIP WG 8.1 Working Conference. North-Holland.

Leong-Hong, B., Plagman, B. (1982). Data Dictionary/Directory Systems. Administration, implementation and usage. John Willey and Sons Inc.

Sovory, S. (ed.) (1988). Artificial Intelligence and Expert Systems. Ellis Horwood Series in Artificial Intelligence. Wiley.

Yurchenko, V. (1987). The functional aproach to automation of interaction systems design. In: Mathematical Methods of Informatics and Software in Systems Modelling. — Collections of papers. Vol. 15. Moscow: The Institute for Systems Studies, pp. 25–35 (in Russian).

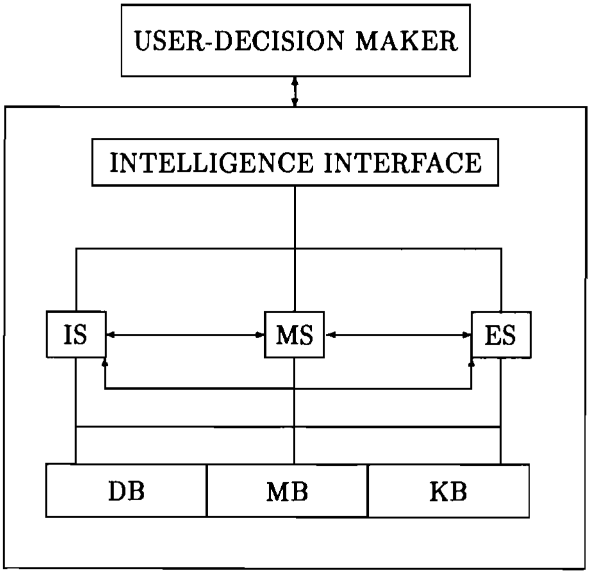


Figure 1: DSS structure

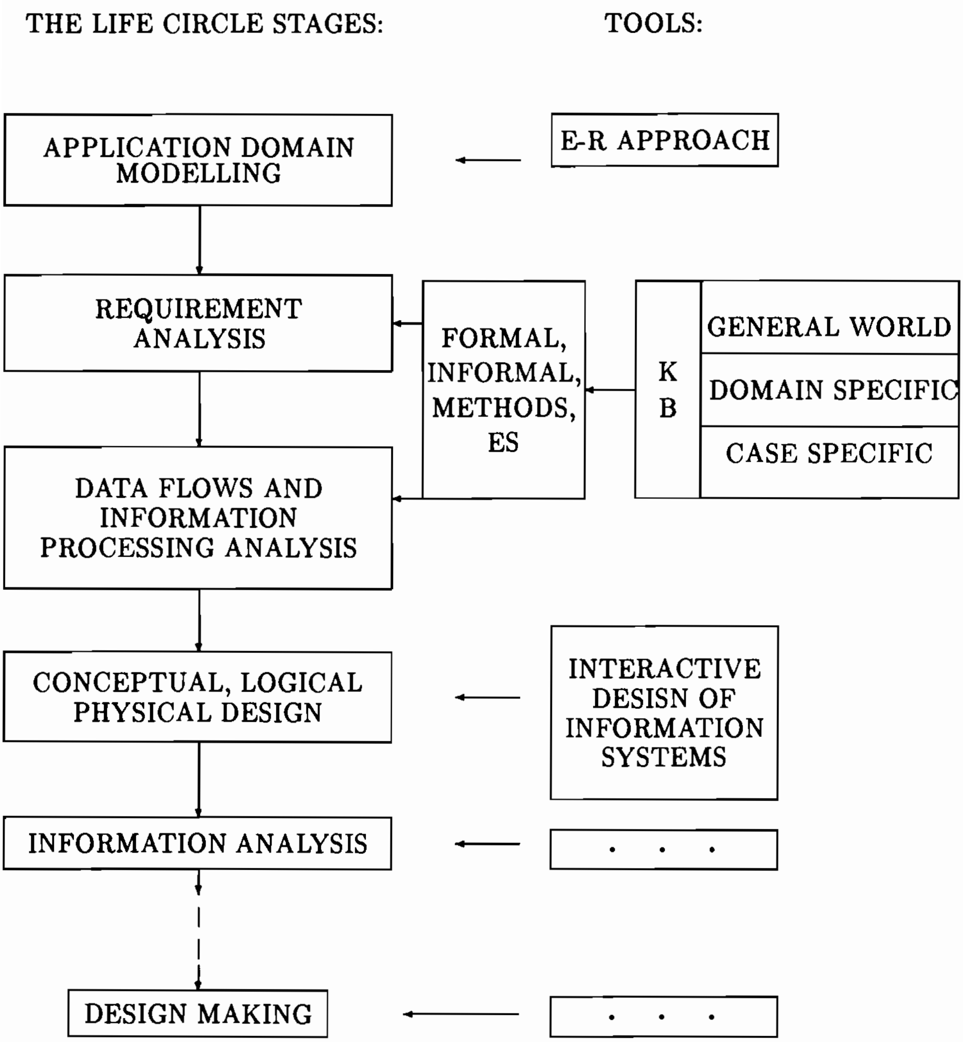


Figure 2: The scheme of interaction of the IS stages and design tools.

An Optimization Framework for a DSS in Programming Development of the Industry

Grzegorz Dobrowolski *
Joint System Research Department
of the Institute for Control and Systems Engineering,
Academy of Mining and Metallurgy, Cracow,
and the Industrial Chemistry Research Institute, Warsaw

Abstract

A problem of programming development of the industry is, in its core, the problem of choice of new technological routes with respect to local conditions and, especially, to their projection into the future.

A naive structuralization of the problem may be: to gather all possible technological proposals, in the sense of a single technological unit, to model their interrelations and finally to organize an optimization problem to choose a recommended alternative.

The paper gives an outline of such an optimization problem and discusses its role in the structure and functionality of Decision Support System designed for supporting development of the chemical industry.

A status of the optimization problem in a DSS is different than in technical applications. The optimization algorithm incorporated in a DSS creates essential view on the modeled system (only some alternatives have been chosen) and opens possibility for evaluation and ordering alternatives based on just few automatically calculated indicators (criteria).

Thus the optimization algorithm serves here as a filter that throws off the majority of detail information and well meets the main idea of a DSS - to help the user to comfortably learn about behavior of the modeled system.

1 Development or Pre-investment Studies. Identification

The project development cycle comprises the pre-investment, the investment and the operational phases.

*This study was partly sponsored by the Ministry of Education, Program RP.I.02.

The pre-investment phase is divisible into several stages: identification of investment opportunities (opportunity studies), preliminary project selection and definition (pre-feasibility studies), project formulation (feasibility studies), the final evaluation and investment decision. There are also support or functional studies (market, raw materials availability, scale and location, etc.) assisting the project formulation stage that are usually carried out by specialized agencies.

Although, it is not so easy to differentiate between an opportunity, a pre-feasibility and a feasibility study because of inaccurate use of these terms in every-day practice some definition will be necessary to link a modeled decision situation with reality. It will be done following results reported in United Nations UNIDO, (1978).

An opportunity study should identify investment opportunities or project ideas, that will be subject to further scrutiny once the proposition has been proved viable with respect to:

- availability of natural resources,
- the future demand for certain consumer goods accordingly to increasing population,
- import to identify areas for import substitution,
- manufacturing sectors successful in other countries with similar local conditions,
- possible interconnections with other industries,
- possible extension of existing lines of manufacture,
- enlarging existing industrial capacities to attain economies of scale,
- the general investment climate and industrial policies,
- cost and availability of production factors,
- export possibilities.

Opportunity studies give rough ($\pm 30\%$) evaluation of alternatives, based on aggregates, than detailed analysis. Data are taken from comparable existing projects and not from quotations of equipment suppliers and the like.

According to an area of interest there are three types of general opportunity studies: area studies (for a region, province, etc.), subsectoral studies (for delimited subsector of an industry) and resource-based studies (for utilization of a given product). Specific project opportunity studies follow identification of general investment opportunities and transform an idea into an investment alternative. The purpose of the studies is to obtain a quick and inexpensive characteristic of investment possibilities.

The next two stages are consecutively more detailed (accurate) and need proportionally more funds assigned to them. But the structure of a pre-feasibility study should be the same as that of a detailed feasibility study. An outline of a pre- and feasibility study should look as follows:

1. Market and plant capacities.

2. Material inputs (approximate input requirements, their present and potential supply positions, and a rough estimate of annual costs of local and foreign material inputs).

3. Location and site (preselection, including, if appropriate, an estimate of the cost of land).

4. Project engineering (technologies and equipment, and civil engineering works).

5. Plant organization and overhead costs.

6. Manpower (requirements and estimated annual cost broken down into labour and staff, and into major categories of skills).

7. Implementation scheduling.

8. Financial and economic evaluation.

As the last point of enumeration has strong relation to the subject of the paper it will be valuable to present majority of ways and indicators used for investment alternative evaluation (Peters and Timmerhaus, 1980). It is assumed that financial calculations are always based on expected market prices of inputs and outputs and are made at the end of each year, and preferably over the lifetime of the project.

From an entrepreneur point of view the main criterion is the financial return on invested capital, that is, the profit. Consequently, the ratio between the profit and the capital invested plays a central role in project evaluation and can be compared with the going rate of interest in the capital market. The full bunch of financial indicators is as follows:

- total investment cost and working capital, fixed assets,
- project financing (capital and financing structure, interest),
- production cost (fixed and variable),
- derivative indicators as: pay-off period, break-even point, simple and internal rate of return.

When it is desired to evaluate the contribution of a development alternative to the national economy [1] , it is necessary to use one of the methods of the cost-benefit analysis developed for the purpose (United Nations UNIDO, 1972). The raising of aggregate consumption is a fundamental objective in this case. Various indicators can be weighted and combined then:

- direct and indirect costs and benefits to aggregate consumption
- shadow prices of labor, foreign exchange and investment,

[1]Reported identification is done for developing countries relying on industrial development planning. For countries without public-sector planning this paragraph is irrelevant as well as some items of opportunity and feasibility study outlines.

- the social rate of discount,
- project exchange rate and foreign exchange savings,
- employment-creation and industrial diversification effect.

2 Subsectoral Development Studies in Chemical Industry

A large proportion of products of the chemical industry are derived from only few raw materials. As the processing of natural resources with mineral or agricultural origin is going downstream, the chains that represent ways of processing are branching from one generation of intermediates to another. To complicate this figure some chains are coupled together across few generations making loops. Moreover, there exist alternative routes leading to particular intermediates and there are also alternative routes and technologies that produce various final products. So the developed chemical industry presents ever changing network of interlinked technologies.

The above short characteristics of the chemical industry drives to a conclusion that development studies in the field ought to be decomposed into at least two levels:

- a higher that corresponds to traditional opportunity and/or pre-feasibility studies simultaneously done for many technological proposals within a chemical branch,
- a lower that corresponds to feasibility studies independently done for technologies preselected on the higher level.

The decomposition (not analyzed here formally) gives a reasonable compromise between a big area of searching defined by intensity of cooperation links (huge amount of data and high cost of the studies) and accuracy of the studies needed for a final decision that will be taken with respect to few alternatives. The higher level generates an investment alternative consisting several technologies and/or its complexes. It is assumed that co-operation links of preselected technologies (complexes) are calculated on that level and analysis of a single investment alternative can be deepened by feasibility study for each technology. The higher level will be considered now.

Technological proposals (production processes) are searched for that ought to be implemented in the existing structure of the branch. The strategy of searching is a choice from among candidates that can be:

1. Different production processes, either for the same final product (these processes, in turn, are influenced by the machinery and equipment, materials and inputs used), or for different types of intermediate, final and by-products.
2. Different scales of production.
3. Different location and sites.

4. Different project implementation scheduling, caused, for example by scarcity of funds.

Criteria of the choice ought to be the same as financial and economic indicators calculated in traditional feasibility study carried out on the lower level to create a substitute of coordination (in terms of formal decomposition).

3 A Sketch of an Optimization Problem

A particular branch of the chemical industry can be modeled as a network of production processes aggregated to simple production functions and distribution flows for a group of chemicals specified. The following components are considered in the model:

- chemicals and other resources and their flows represented by balance nodes satisfying the mass balance principle,
- chemical transformations represented by processes that process chemicals exchangeable with the environment into other exchangeable chemicals,
- other indispensable resources also in terms of their flows.

A simple transition function is defined for the process elements by yield and consumption coefficients together with a capacity constraint. The network is constructed in the way that processes are connected to each other throughout balance nodes.

Relations between the components of the model are to reflect well-known phenomena existing in the chemical industry:

- a chemical can either go to or be obtained from several transformations,
- a chemical transformation may either produce or consume several chemicals,
- one or more chemical processes can run on one chemical installation,
- certain resources are necessary for operation of any installation,
- chemicals and resources can be exchanged via their flows with the environment.

Interaction with the environment is assumed by I/O flows of chemicals and other resources (sources and sinks of the network).

To reflect the possibility of running several different chemical processes on the same hardware the element called installation was introduced. Processes of the particular installation are dependent on a common capacity constraint.

The idea of installation splits I/O flows into two categories called:

media - flows that are interchanged between processes themselves and between processes and environment, i.e. raw materials, products and by-products.

special resources - flows that are interchanged between installations and environment (common for all processes running on a particular installation), e.g.. investment, manpower.

Nobody ought to be surprised that a media concept is for modeling not only particular chemicals, energy carriers but also pollutants in bulk (solid waste, liquid waste, emission) or in groups of toxic substances.

Formulas of the model will be written in dynamic, discrete manner with an independent variable $t = 0, 1, \ldots, T$ where T is a given time horizon.
There are following types of elements of the model:

- Installations - indexed by set I,
- Processes - indexed by set P,
- Media - indexed by set J,
- Markets - indexed by set M,
- Special resources.

Data about a process ($i \in I,\ p \in P$) are represented in our description by two functions:

$q_i(t, t_p)$ investment (fixed assets) and other fixed cost distributed over time,

$\bar{z}_p(t, t_p)$ capacity of a process p over a life-time.

The functions are bounded by the same moment of time t_p, namely time of implementation of a process p. A set of all $\{t_p\ ,\ p \in P\}$ represents an investment alternative giving a list of processes to be built and a schedule of implementation.
Let us introduce variables of the model:

$z_p(t),\ p \in P$ - a level of production of process p over time $t = 0, 1, \ldots, T$,
$y_j^m(t),\ j \in J,\ m \in M$ - an amount of medium j bought or sold on market m over time $t = 0, 1, \ldots, T$.

An equation describes amount of a given medium produced or consumed.

$$y_j(t) = \sum_{p \in P_j^+} b_{jp}\ z_p(t) \ - \sum_{p \in P_j^-} a_{jp}\ z_p(t) \quad , \quad j \in J \tag{1}$$

while the symbols above are defined as follows:
P_j^- - a subset of processes where medium j is consumed,
$a_{jp}\ z_p(t)$ - quantity of medium j consumed (consumption coefficient),
P_j^+ - a subset of processes where medium j is produced,
$b_{jp}\ z_p(t)$ - quantity of medium j produced (yield coefficient).

It may be immediately assumed that yield and consumption coefficients are constant for our purpose.
A balance equation for media is:

$$\sum_{m \in M_j^+} y_j^m(t) \ - \sum_{m \in M_j^-} y_j^m(t) = y_j(t) \quad , \quad j \in J \tag{2}$$

where:

M_j^+ - a subset of markets where medium j may be sold,
M_j^- - a subset of markets where medium j may be bought,

Market constraints are represented by:

$$\underline{y}_j^m(t) \leq y_j^m(t) \leq \bar{y}_j^m(t) \ , \quad m \in M_j^+ \cup M_j^- \ , \quad j \in J \tag{3}$$

while capacity constraints for processes by:

$$\sum_{p \in P^i} \frac{1}{\bar{z}_p(t, t_p)} z_p(t) \leq 1 \ , \quad i \in I \tag{4}$$

where: $\bar{z}_p$ - production capacity of the process p.

When several processes are to run on the same installation, the capacity is calculated under an assumption that the particular process occupies the whole installation.

On the above base, financial and economic evaluation for the whole network can be done. A formula for net present value is reproduced here:

$$\sum_{t=0,\dots,T} \frac{1}{(1+\alpha)^t} \left[\sum_{i \in I} \sum_{p \in P^i} q_i(t, t_p) + \sum_{j \in J} \left(\sum_{m \in M_j^+} c_j^m(t)\, y_j^m(t) - \sum_{m \in M_j^-} c_j^m(t)\, y_j^m(t) \right) \right] \tag{5}$$

where:

c_j^m - a price of medium j on the market m,
α - an assumed discount rate.

The net present value is known as the best representation of profit. It is counted over the whole lifetime of the investment alternative and a changing value of money is considered by applying a discounting procedure. It is obvious that other economic indicators enumerated in the paper can be easily defined in terms of the model.

Under detailed assumption with respect to the functions: $q_i(t, t_p)$, $z_p(t, t_p)$, $i \in I$, $p \in P$ and, especially, to a way of economic evaluation, a simple or multi objective, dynamic or static optimization problem can be entirely defined and solved. Details of the model stay beyond the scope of the paper but play an important role in DSS building and even decide about its efficiency. A reader is referred to two operational variants of the model: a static one (Dobrowolski and Zebrowski, 1988) that constitutes a core of the Multiobjective Interactive Decision Aid in development of the chemical industry (Dobrowolski et al., 1984; Dobrowolski and Rys, 1988) and a dynamic one (Dobrowolski and Rys, 1982) that has been not implemented so far.

4 Decision Support with Optimization. Conclusion

Incorporation of optimization calculations into a DSS adds new aspects to its architecture and functionality (Dobrowolski and Rys, 1988). An optimization subsystem arises that must be integrated with the rest of the system. It can be done by introducing following elements:

- a generator of an optimization problem, using data mainly taken from a data base of the system,
- a solver that performs calculations,
- a module that maintains appropriate references of optimization results to a rest of data, e.g. by loading them into relational structure of a data base,
- a postoptimal analysis module that tightens better parameters of the model with a solution to the optimization problem.

To sustain the main idea of a DSS, it is supporting decision (not generate it as in expert systems), the optimization subsystem has to offer, especially created, user-interface that enables for interactive modification of the optimization problem in the sense of its parameters and structure. Alterations ought to be done by changes of the input data that have a direct and clear interpretation to a user.

Although there are some devices that help the user to converse with the system, his position becomes a little bit harder than in the situation not supporting by optimization calculations. In a chain of data processing a new element—optimization algorithm arises that can be observed by the user as not understandable jump on the way towards a final decision. More serious danger is imposed by applying a multiobjective optimization algorithm that is frequently not fully controlled or observed by the user. A good illustration of a new status of the user can be an extended decision sphere that now deals with:

1. Applying a DSS and especially that with an optimization core.
2. Accuracy of input data taken.
3. Strategy of choice (criteria for the optimization problem).
4. Analysis of alternatives having in mind their origin from the optimization problem.
5. Discarding alternatives and a final choice (decision).

But advantages are much more important. An existing optimization model stands for a base that structuralizes the whole approach, induces a structure and a procedure of a DSS and makes user's activity more disciplined. Hard calculations are done automatically by a solver and it is not necessary to start them in the system or even carry them out outside.

A status of the optimization problem in a DSS is different than in technical applications. Let us remind that accuracy of data for the opportunity study allows only for $\pm 30\%$ approximation of main economic indicators used here as criteria. Usually, several solutions exist that are indifferent due to the accuracy of data. A remedium from such a situation is resignation of optimality for the sake of acceptability. A postoptimal analysis module and a reach set of results reviews can help the user.

The optimization algorithm incorporated in a DSS creates essential view on the modeled system (only some alternatives have been chosen) and opens possibility for evaluation and ordering alternatives based on just few automatically calculated indicators (criteria).

Thus the optimization algorithm serves here as a filter that throws off the majority of detail information and well meets the main idea of a DSS - to help the user to comfortably learn about behavior of the modeled system.

References

Dobrowolski, G., Kopytowski, J., Wojtania, J. and Zebrowski, M. (1984). Alternative Routes from Fossil Resources to Chemical Feedstocks, IIASA Research Report RR-84-19, International Institute for Applied System Analysis, Laxenburg, Austria.

Dobrowolski, G. and Rys, T. (1982). Dynamiczny model programowania rozwoju sieci produkcyjnohandlowej Teoria i aplikacja. In Symulacja systemow gospodarczych. Szkola Trzebieszowice 81. Supplement, Nowak, K. and Pelech, A., eds., Towarzystwo Naukowe Organizacji i Kierowania, Wroclaw, Gliwice, 1-22.

Dobrowolski, G. and Rys, T. (1988). Architecture and Functionality of MIDA. In Theory, Software and Test Examples for Decision Support System, Wierzbicki, A. P. and Lewandowski, A. eds., IIASA Working Paper WP-88-071, International Institute for Applied System Analysis, Laxenburg Austria.

Dobrowolski, G. and Zebrowski, M. (1988). Basic Model of an Industrial Structure. In Theory, Software and Test Examples for Decision Support System, Wierzbicki, A. P. and Lewandowski, A., eds., IIASA Working Paper WP-88-071, International Institute for Applied System Analysis Laxenburg, Austria.

Peters, M. S. and Timmerhaus, K. D. (1980). Plant Design and Economics for Chemical Engineers, McGraw-Hill, Inc., 1980, ISBN 0-07-049582-3.

Thuesen, G. J. and Fabrycky, W. J. (1984). Engineering Economy , Prentice-Hall, Inc., ISBN 0-13-277723-1.

United Nations UNIDO (1972). Guidelines for Project Evaluation, United Nations Publication #E.72.II.B.11, Vienna, New York.

United Nations UNIDO (1978). Manual for the Preparation of Industrial Feasibility Studies, United Nations Publication #ID/206 #E.78.II.B.5, Vienna, New York.

On the Role of Software in Interactive Decision Analysis

Raimo P. Hämäläinen and Risto Karjalajnen
Systems Analysis Laboratory
Helsinki University of Technology
Otakaari 1 M, 02150 Espoo, Finland

Abstract

This paper deals with computer assisted decision analysis, illustrated by a real-world project in the field of risk analysis. The aim is to analyze the role of computerized aids in different phases of the interactive decision analysis. The presentation follows three parallel lines. First, the process of decision analysis will be introduced. Second, the use of software within the process will be illustrated by describing a recent project. A decision analytic approach was introduced to the members of the national energy committee, studying risks and long-term effects of different alternatives for energy production. Third, the factors affecting the role of software will be discussed in connection with the respective phases of the analysis.

1 Introduction

The aim of the present paper is to analyze the benefits of the use of computerized aids in decision analysis. Here, decision analysis refers to an approach supporting the decision making process, rather than to a unified methodological or theoretical approach. Often, the most important gains arise from the learning process that takes place during the analysis. The usefulness of the approach or any particular method cannot, however, be proven except through real life applications. Use of computerized aids is essential in the practice of decision analysis. Nevertheless, little attention has so far been paid to the aspects of computer implementations. There are only few commercial software products available (John, 1987), reflecting the relative infancy of the field. The existing software tools have in most cases been developed to meet the needs of researchers.

Several methods of decision analysis have emerged since the pioneering works of von Neumann and Morgenstern in the early 1950's. The original formulation of the expected utility theory provided an essentially normative basis for decision making. However, the human behaviour in real situations has been found to deviate from the theoretically predictable manner of making decisions, as demonstrated through several paradoxes (see e.g. Allais and Hagen, 1979). As a consequence, various alternative theories or generalisations have been proposed. This include the prospect theory (Kahneman and Tversky, 1979) and the Analytic Hierarchy Process (Saaty, 1980). Moreover, there are formulations accounting for e.g. disappointment (Bell, 1985) and regret (Bell, 1982, 1983) of a decision maker (DM). Recently, heuristics and biases affecting the decision making process have been described (Kahneman et al., 1982). Behavioural research may in time lead to an increased ability to describe the decision making process. The evolving formulations require empirical as well as theoretical justification, leading to the need of experimentations with the proposed methods. The role of software will be of growing importance as the newly developed methods are also being applied to real-life decision situations.

The process of decision analysis

In spite of the above shortcomings, decision analytic methods have been successfully applied to various real-world problems (Keeney and Raiffa, 1982; Watson and Buede, 1987; von Winterfeldt and Edwards, 1986; Zahedi, 1986). The process of decision analysis may be divided into different steps (see e.g. EPRI, 1987). However, in practice, the analysis proceeds in an iterative manner, which makes it difficult to make a clear-cut distinction between the following phases:

- *Structuring the decision problem.* This involves the definition of the decision context, i.e. identifying the fundamental objectives of the DM, the feasible alternatives and the criteria used to compare the alternative solutions.
- *Assessment of consequences.* The possible consequences of proposed alternatives are estimated. The intangible nature of impacts, long time horizons involved and the interdisciplinary skills required are some of the factors that may pose difficulties.
- *Determining the preferences of a DM.* The problem is analyzed from the subjective point of view of a DM, based on his or her values. The preference model is a means to make the relevant values explicit in the context in question.
- *Sensitivity analysis.* The goal of sensitivity analysis is to identify those factors or assumptions, that the solution is most sensitive to. At its best, this vital part of decision analysis is conducted with respect to the problem formulation, the consequences of the alternatives and the subjective values of the DM.

The aim is to decompose the analyzed problem into smaller elements, which may be combined according to the chosen method. The analysis may be viewed as a learning process,the aim of which is to clarify the relevant factors to be considered. Decision analysis offers a framework where expert opinions may be combined with objectives and values of a DM. Especially in public sector decisions there are several DM's and other interested parties with different concerns. The decision analytic framework may be used to facilitate communication and negotiation between the different parties involved in the decision process (Keeney, 1988).

2 Illustrating the use of software

The use of software will be illustrated by a description of a real-life decision analysis project dealing with risks related to the energy production. Public energy policy is a typical application field of decision analysis. There are several actors on the field, such as governmental agencies, concerned citizen groups and energy production companies. The possible consequences of the energy alternatives reach far in the society and in the future, and there is quite a lot of uncertainty involved. The government of Finland appointed a committee in 1987 to study the societal and environmental effects and risks of different energy alternatives, as put in the memorandum enclosed with the appointment (Council of State, 1987):

> ...The task of the committee would be to appraise the available information about the societal and environmental effects and risks of the different energy alternatives, as well as to attempt to sketch a more general context to take these points of view into consideration in the energy policy decision making in the long term. Should the committee not regard it possible to take a stand in the societal acceptability of different energy alternatives, it should in any case attempt to make an organized proposition of the points of view that should be taken into consideration when assessing the acceptability of the different energy alternatives...

The committee members are experts from various fields, usually in a high academic standing. Eleven members of the committee participated in the risk analysis project. Decision analysis was applied to the assessment of risks and societal impacts of the energy alternatives. The working sessions were organized separately with each participant during June 1988. The project served the committee members as an introduction to the decision analytic approach. The manner of conducting the computer assisted analysis will be briefly described below.

The analysis in practice

In the risk analysis project, the problem structuring was no simple task, as there was no *a priori* defined decision context. The participants were therefore asked to formulate a decision situation. The formulations tended to follow a pattern of long term planning of the Finnish energy policy, the time horizon extending somewhat after the year 2000. The candidate alternatives included options such as the fifth nuclear power plant, coal-fired power plants, the use of natural gas and the conservation of energy.

The objectives were elicited during a discussion between the analyst and the participant. The fundamental objectives were organized in a form of a *value tree* (EPRI, 1987). It is a hierarchy, where the most specific factors are located on the lowest levels. Typically, factors such as impacts on health and the environmental changes were included in the highest level of objectives. Proceeding to lower levels, impacts on health included factors such as injuries, the catastrophe risk and the risk of long-term impacts, e.g. genetical changes. Pollution and climatical changes were among the factors often included in the environmental changes. Factors such as the public concern, equity and the acceptability of decision processes were also mentioned.

No aids except paper, a pencil and a flapboard had been utilized so far. At this stage the completed value trees were reconstructed with TREEVAL, a software product developed at the Department of Systems Science at the University of Southern California.

The participants were experts in the fields included in the scope of the committee. However, estimation of consequences of the energy alternatives was seen as a difficult task. This was probably due to the broad range of impacts and the inability to identify the effects of energy policy planning among the other factors. Thus, the assessment was largely a matter of judgment, as recognized by the participants. Numerical measurement scales, i.e. attributes, were constructed for each of the lowest level objectives, and the impact of the alternatives was estimated on these scales.

The project involved a construction of a simple preference model (EPRI, 1987; von Winterfeldt and Edwards, 1986). TREEVAL was applied to the assessment of an additive value function. The procedure involved the assessment of unidimensional value functions for each of the attributes. These were combined as a weighted sum, yielding the additive value function. The results of the model corresponded to the acceptability of the different decision alternatives, according to the subjective value judgments of the participants. Figure 1 illustrates the way of representing the value tree with TREEVAL.

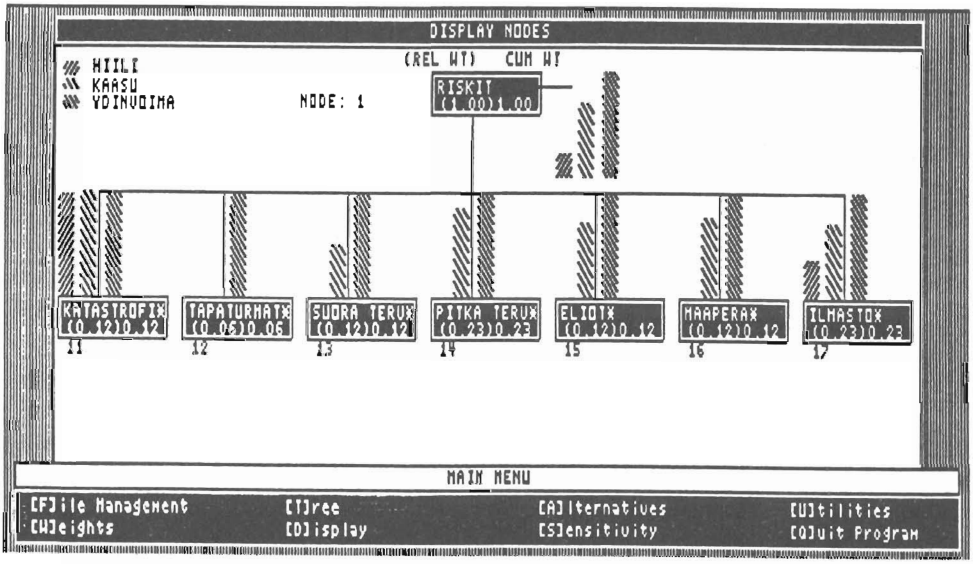


Figure 1: A value tree.

The problem was further analyzed, taking into account the social and economic impacts of the decisions. The energy decision model, used before in a national energy debate in the Parliament of Finland (Hämäläinen et al., 1985; Hämäläinen, 1988), was the starting point in this case. The Analytic Hierarchy Process was applied to the construction of the preference model. The procedure was aided by HIPRE, developed at the Systems Analysis Laboratory. The relative importance of the lowest level criteria may be represented graphically by HIPRE. Illustration in Figure 2 is extracted from a working session with one of the participants. The division of weights among the alternatives is shown in the Figure as well. The composition of the alternative weights from different decision criteria is illustrated in Figure 3; this is just another way of looking at the same results.

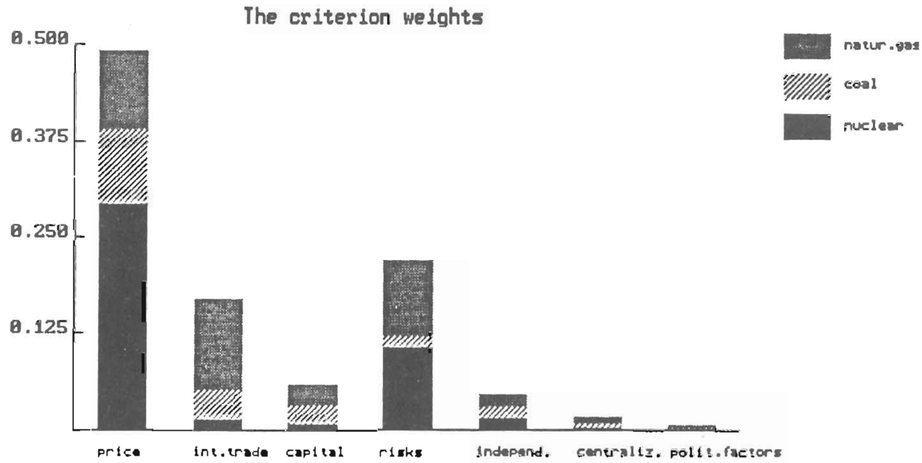


Figure 2: Weights of the decision criteria.

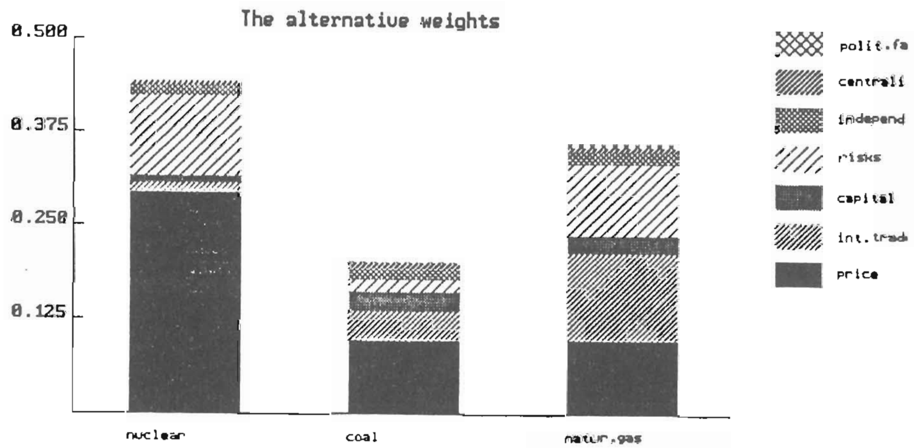


Figure 3: Weights of the alternative energy policies.

HIPRE was also applied to the sensitivity analysis. Figure 4 shows graphically the changes in the weights of the alternatives versus the change in the weight of a decision criterion. The participants of the risk analysis project found such a visual presentation of the sensitivity analysis one of the most interesting aspects of computer assisted decision support.

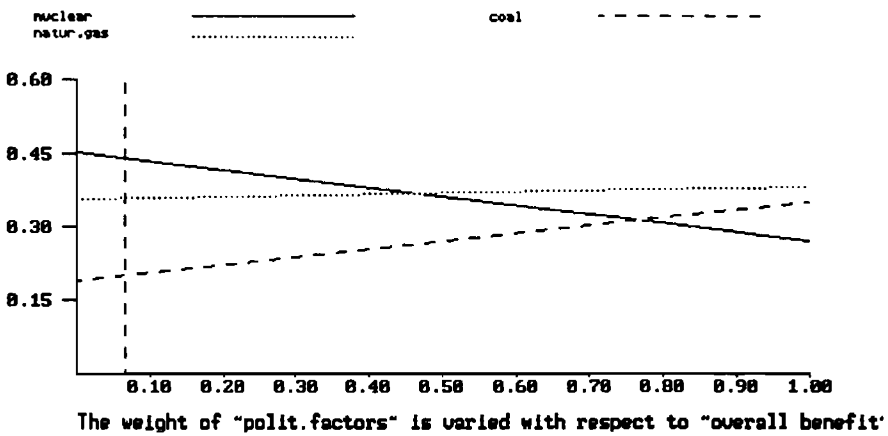


Figure 4: A sensitivity analysis.

3 The role of software

The present supply of the decision analytic software offers aids to various phases of the analysis. Most of the products available are part of the so-called second generation of the decision analysis software (John, 1987). The overall manner in which the aids are applied differs from the multicriteria optimization. Computational requirements remain rather moderate throughout the analysis. The role of software may thus be extended beyond that of a computational tool. In the decision analytic context, the software may have a value of its own. The software may provide a guideline, making the analysis proceed smoothly from one phase to another. It may also help the DM to concentrate on the analysis, ensuring the intensity of the interactive process. The actual usage of computerized aids varies according to the phases of the analysis.

Problem structuring

Problem structuring remains more an art than a science. The techniques include hierarchical modelling, construction of decision trees, network modelling, building of structural models or influence diagrams and goal-directed structuring. The common thread is the aim to encourage creativity in the initial problem formulation, in order to avoid a narrow interpretation of the decision context.

Software products supporting this phase remain few, even though option generation remains a problem. In complex decision situations it may be difficult to come up with even few feasible alternatives. The opposite case in the one where screening is necessary to prune the alternative set to a manageable size. Rule-based expert systems may be of use, such as the one developed by Ruusunen and Hämäläinen (1988) for evaluation of R&D projects. In case of several DMs, analysis may be carried out with different groups, and the formulations may be combined to yield an overall view of the decision. This poses a need for a negotiation support software, which to our knowledge has scarcely been met so far.

There are few aids for deduction of a value tree. This should come as no suprise considering the amount of human creativity needed to make the an explicit description of DM's objectives. Instead, there are several software products supporting the value tree construction, including HIPRE, TREEVAL and Expert Choice (by Decision Support Software, Inc). VALPRE, developed at the Systems Analysis Laboratory for problem structuring and preference analysis, provides an example of a visualization of a value tree (Figure 5).

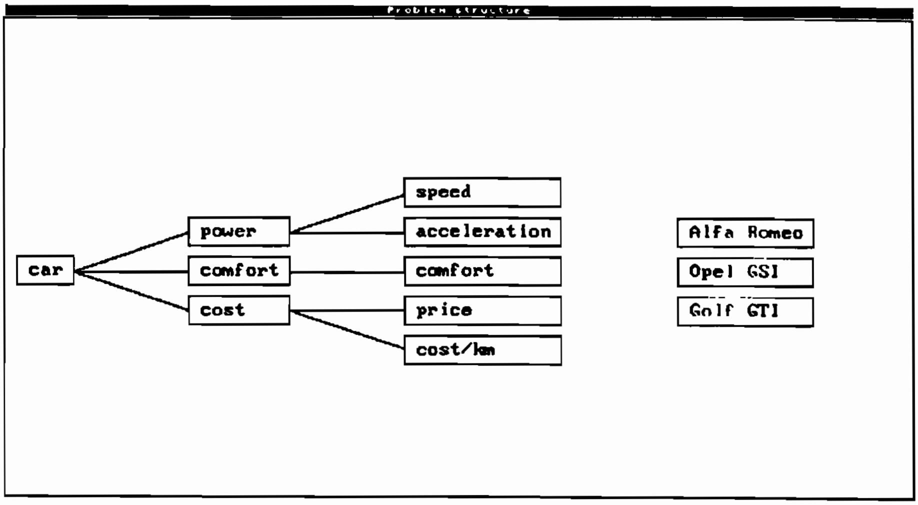


Figure 5: Visualization of a problem structure.

Assessment of consequences

The ability to accurately assess the consequences of decision alternatives is influenced by numerous factors. These include the strategic nature of decisions, the possibly long time horizons involved and the indirect societal impacts. In addition to the practical difficulties in gathering the information, appraisal of the available knowledge requires human judgment, leading to inherently subjective estimates. Biases may occur in the estimates, due to the often misleading heuristics (Kahneman et al., 1982). Thus, the challenge is to aid decision making, knowing the deficiencies in the assessment process. The eliciting and combining of expert judgments requires calibration and debiasing procedures, for which purposes there are few, if any, software aids available. Visual feedback can be provided to facilitate the input of estimates in the preference model, as implemented in VALPRE (Figure 6).

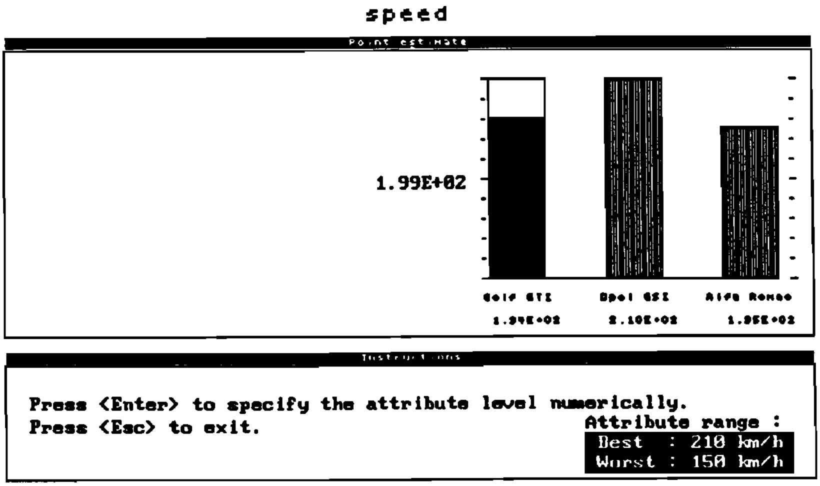


Figure 6: Specifying the impacts of the decision alternatives.

Determining preferences

The computer supported construction of a preference model encourages learning in the interactive decision analysis. The eliciting of the preferences may be facilitated by the use of graphics. Figure 7 shows an example from VALPRE, applied to a construction of a unidimensional value function. As the results provide immediate feedback, the DM is motivated to iterate between the different phases of the analysis, gaining a better understanding of the problem in question.

On the other hand, the construction of a numerical model may lead to an illusion of accuracy. This is especially the case if the consequences and relations between objectives are hard to measure precisely. Most of the related commercial products are implementations of relatively simple preference models. As the simplicity facilitates the applications, it carries the danger of loss of credibility with it, as well. The third generation software should indeed offer means to test the correctness of assumptions regarding the preference model structure (John, 1987).

Presentation of the preference model to a DM deserves special attention. The weights of objectives in an additive value function reflect scaling of different units as well as the priorization of the objectives. This may lead to potentionally counterintuitive results, as a major concern of the DM may be assessed a minor weight in the preference model. The difficulty in explaining such features to the DM may well cause a loss of credibility.

Sensitivity analysis

The sensitivity of the results may be calculated with respect to the initial assumptions, the data used or weighting of the objectives. Confidence in the validity of the analysis may be strengthened by another analysis conducted in a different matter. Still, a positive

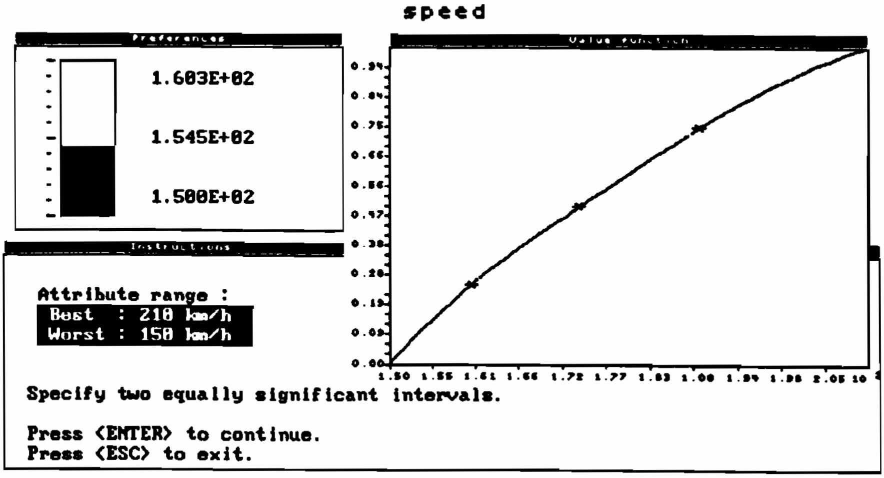


Figure 7: Constructing a unidimensional value function.

correlation between the results from alternative approaches may imply nothing more than the existence of common underlying assumptions or prejudices. In practice, the software implementations such as TREEVAL or Expert Choice offer an option to sensitivity analysis with respect to the data used or the preferences. The resulting sensitivity is usually shown graphically, as with HIPRE in the risk analysis project (Figure 4). The effectiveness of such a visual feedback justifies well the effort of automation of the procedures involved. In the future artificial intelligence may offer more sophisticated means to find the highly sensitive assumptions made in the analysis (John, 1987).

Future trends

A recent project (Salo and Hämäläinen, 1988) points out some future directions in the development of decision support systems. The system supports financial planning of telecommunications network investments, and it is currently in active use in the Finnish Post and Telecommunications Agency. The system was implemented on an artificial intelligence workstation. The software is menu-driven, handled by the use of a mouse as a pointing device. Graphical aids, similar to the input gauges in Figure 6, simplify the interaction with the user. The use of windows and separate processes makes the environment a flexible one. It may also be viewed as a collection of tools. The user is left a choice which of them to actually utilize, according to the problem. The ability to adapt to different situations goes parallel with the different strategic nature of decision problems, varying in the amount of detail.

There is a growing pressure towards standardization within the computer industry. One particular consequence is the availability of systems for management of the software interfaces. The goal is the separation of the application and the user interfaces. The interaction with the user is usually based on the use of menus and pointing devices. The specification of the user interface with so-called window managers preserves the functional and visual appearance of a software product intact when transferred to different environments.

4 Conclusion

The role of software in different phases of the decision analysis is affected by numerous factors. The successfully integration of the software products into the process requires experience of real-life applications. The practical experience provides also information about the nature of decision situations into which the particular methods can be adapted. The recent development of more user-friendly — and more effective — software engineering environments facilitates the creation of computerized decision aids. The availability of more sophisticated tools for practice of decision analysis points out the potential to extend the role of software beyond that of a computational aid. Ease of use and the immediate feedback provided by the interactive software encourage the decision maker to gain insight of the problem in question. The software can thus become an essential part of the decision solving environment.

References

Allais, M. and Hagen, O., eds. (1979). Expected Utility Hypothesis and the Allais Paradox. D. Reidel Publishing Company, Dordrecht, 714p.

Bell, D.E. (1982). Regret in Decision Making under Uncertainty. *Operations Research*, Vol. 30, pp. 961–981

Bell, D.E. (1983). Risk Premiums for Decision Regret. *Management Science*, 29, pp. 1156–1166

Bell, D.E. (1985). Disappointment in Decision Making under Uncertainty. *Operations Research*, Vol. 33, pp. 1–27

Council of State (1987). Decision No. 43/410/87 of the Council of State of Finland.

EPRI (1987). Operational Procedures to Evaluate Decision With Multiple Objectives. Electric Power Research Institute, EA-5433.

Hämäläinen, R.P. (1988). Computer assisted energy policy analysis in the parliament of Finland. *Interfaces*, Vol. 18, No. 4, pp. 12–35.

Hämäläinen, R.P., Oehlandt, K., Seppäläinen, T. and Ruusunen, J. (1985). An approach to decision support in national energy policy planning. In: Proceedings of the IEEE Conference on Systems, Man and Cybernetics, Tucson, pp. 1008–1012

John, R.S. (1987). Multiattribute Utility Analysis Software Design. A manuscript of the presentation held at the ORSA/TIMS conference, St. Louis, October 26.

Kahneman, D. and Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, Vol. 47, No. 2, pp. 263–292.

Kahneman, D., Slovic, P. and Tversky, A., eds., (1982). Judgment under uncertainty: Heuristics and biases. Cambridge University Press, 555p.

Keeney, R.L. (1988). Structuring objectives for problems of public interest. *Operations Research*, Vol. 36, No. 3, pp. 396–405

Keeney, R.L. and Raiffa, H. (1982). Decision with Multiple Objectives Preferences and Value Tradeoffs. Wiley, 569p.

Ruusunen, J. and Hämäläinen, R.P. (1988). Project selection by an integrated decision aid. A manuscript.

Saaty, T. (1980). The Analytic Hierarchy Process. McGraw-Hill, New York, 287p.

Salo, A. and Hämäläinen, R.P. (1988). SETELI — The Strategy Expert for Telecommunications Investments. VHS/PAL/SECAM/NTS video tape, 17 min, Helsinki University of Technology, Systems Analysis Laboratory.

Watson, R.S. and Buede, D.M. (1987). Decision synthesis: the principles and practice of decision analysis. Cambridge University Press, 299p.

von Winterfeldt, D. and Edwards, W. (1986). Decision analysis and behavioural research. Cambridge University Press, 604p.

Zahedi, F. (1986). The analytic hierarchy process — a survey of the method and its applications. *Interfaces*, Vol. 16, No. 4, pp. 96–108.

Decision Support Systems: An Approach to Facet Classification

Alexey B. Petrovsky, Mikhail Yu. Sternin, Vladimir K. Morgoev
Institute for Systems Studies
USSR Academy of Sciences
Moscow, USSR

Abstract

The following aspects are suggested as a basis for DSS classification: types of system concepts; users; decision making problems; software; areas of practical applications.

1 Introduction

Decision Support Systems (DSS) are a new class of man-machine systems where a user generates and makes decisions by interacting with computer that processes bulks of various information. DSS has been resulted from an evolution and development of Decision Theory, Management Information Systems (MIS) and Data Base Management Systems in the direction of more practical assisting to users in their professional activities.

According to M. Ginzberg and E. Stohr (1982) the DSS concept is shifting gradually from explicit statement "what a DSS does" to operational characteristics "how the DSS's objective can be accomplished". The analysis of different DSS concepts allowed us to suggest the following aspects as a basis for DSS classification: types of system concepts; users; decision making problems; software; areas of practical applications.

2 Types of system concepts

From a conceptual point of view on DSS one can consider an information approach, knowledge-based approach, decision-based approach, tools-based approach.

The conceptual DSS model suggested by R. Sprague (1980) has all the features of the information approach. The general components of this model are "user-system" interface, data base and model base. The "user-system" interface provides user's interaction with data base and model base. Interface includes tools for a dialogue generation and management, data and model management. The ability to formulate decision models and combine them with data bases is one of DSS peculiarities. Simulation techniques have to provide a flexible construction of models from separate blocks and subroutines, and to allow simple procedures of model management.

Evolutionary DSS of R. Bellew (1985) is a generalization of Sprague's model. It has supplemental components such as text base and rule base which contain less structured information (texts in natural language) and more structured information (heuristics, knowledge representation rules).

An essential characteristic of knowledge-based DSS is an ability of "problem recognition" that explicates a new aspect of decision support. Generic knowledge-based DSS suggested by R. Bonczek, C. Hollsapple and A. Whinston (1981) consists of three interacting components: language system, knowledge system and problem processing system.

Language system provides communication between a user and DSS component. Knowledge system contains information on problem field. Knowledge systems are distinguished by data structure and methods of knowledge representation and organisation. Problem processing system or problem processor percepts a problem description prepared in accordance with syntax of language system and making use of models and knowledge transform problem formulation in detailed procedural specification in order to solve the problem. In more sophisticated situations problem processor should be able to create models for the problem solution. According to A. Bosman (1982) the problem processing system of extended generic DSS should also support validation, design and implementation phases of decision process.

The decision-based approach to DSS concept was suggested by O. Larichev and A. Petrovsky (1987). The main components of a decision-model construction, data and model management. The blocks of problem analysis and decision-making incorporate procedures and techniques which help formulate the problem, analyze approaches to its solution, and generate the result by making use of the data, model and knowledge bases. In order to perform its functions the decision making block must contain a library of decision methods including those for solution of multicriteria and single criterion problems on objective and subjective models. Apart from the library this block should incorporate a set of rules or an expert system, adjusted by an experienced consultant on the problem area, permitting formulation of requirements to decision techniques, models and knowledge, selection of the most adequate tools for the problem solution.

Tools-based approach reflects a growing attention to problems of DSS design. R. Sprague proposed to distinguish three levels of systems: specific DSS, DSS generators, DSS tools (Sprague, 1980). Specific DSS are applied by decision makers or end-users to solve their professional decisions. DSS generators are software packages whose capabilities are used by DSS builders to create new specific systems corresponding to user's needs. According to M. Ginzberg and E. Stohr (1982) the DSS generator is to consist of user interface, language interface, data base management system, model management system, data extraction system, system directory. DSS tools are used by technicians or "toolsmiths" who design DSS generators and specific DSS. DSS tools include programming languages, operational systems, input-output facilities, and so on.

3 System users

User-oriented DSS classification includes: a hierarchy of users in management systems; a pattern of system usage; an interdependence of decision makers.

There are high, middle, and low managerial levels. The most popular point of view is that DSS users are top officials. But we accept M. Ginzberg and E. Stohr's opinion (Ginzberg and Stohr, 1982) that a decision maker is a person analyzing alternatives and influencing choice. So a real decision maker can be at a high, middle and/or low managerial level. He can be a manager, an expert, an analyst, a clerk and so on.

S. Alter (1980) suggested that DSS should be classified by the following patterns of system usage: terminal mode (on-line), clerk mode (off-line), intermediary mode, subscription (automated) mode. These patterns of user's interaction with DSS are distinguished by means of receiving information from the system.

P. Keen (1980) divides DSS into systems of personal, group, and organizational support based on a different degree of user's dependence in decision process. Personal Support Systems are intended for support of individual managerial decisions when persons can make decisions self-sufficiently. Group Support Systems provide support of negotiations among several decision makers that require substantial discussions and communications. Organizational Support Systems are used for support of organizational processes with sequential interdependencies between decision makers.

4 Decision making problems

Facet classification of decision making problems was grounded by O. Larichev (1985). These problems are classified by their novelty (unique decisions, recurrent decisions); representation (holistic choice, multicriteria choice); type of model (objective model based on formal regularities, subjective model based on decision maker's preferences). Cartesian product of dihotomies defines eight classes of decision problems, and each one corresponding to specific DSS. Multicriteria decision problems with objective models are the most typical applications of DSS.

The similar DSS classification based on a repetition of decision situations was suggested by J. Donovan and S. Madnik (1977). There are institutional systems and ad hoc systems. The former is applied for solving reccurent decision problems which can be formalized and detailed beforehand. The latter deals with unique decision problems and unforeseen situations.

5 Software

DSS are differentiated also by their software. Depending on software functions S. Alter (1980) identifies data-oriented systems and model-oriented systems. Data-oriented systems perform data retrieval, data processing and/or data analysis. Model-oriented systems provide computations, simulations and/or optimizations of results making use of computation models, representational models, optimization models, and suggestion models. Alter's classification is connected with the information concept of DSS.

A classification scheme of R. Bonczek et al. (1981) is based on the different level of procedurality for "man-machine" interface languages. There are a language handling data retrieval and a language handling computations. Each language may have one of three levels, i.e. a procedural, command or nonprocedural level. A language of procedural level demands detailed descriptions of retrieval procedures or computations. A language of command level works with message generators or models defined beforehand. A language of nonprocedural level permits user to formulate needed data and results in general form. Cartesian product of language procedurality levels gives nine possible DSS classes. Interface languages of present-day DSS tend to move from procedural level to nonprocedural level.

6 Areas of application

Finally DSS can be classified by their practical applications. DSS are distinguished by areas of professional activities: micro economics, macro economics, office automation, technology assessment, medicine, etc. Problems of planning and forecasting in business from small firms to large corporation are the most broad field of DSS applications. Making use of DSS in office automation improves an efficiency of decision and communication processes in organizations. Medical DSS based on knowledge of experienced physicians can assist to diagnose a disease and to choose the optimal way of its treatment.

By analogy with R. Anthony classification of organizational decisions (Anthony, 1965) one can divide DSS by time horizon into systems for strategic planning (long-term decisions), management control (middle-term decisions), operational control (short-term decisions). A lot of authors believe that strategic planning and management control are only fields of DSS applications, and that MIS are more suitable for solving well structured tasks of operational control. But in other papers it is pointed out that DSS are helpful for solving ill-structured problems as in long-term so short-term decisions.

7 Conclusions

In our opinion there are the following trends of DSS development:

- an integration of DSS, MIS, and communication systems;
- a rapprochment of DSS with Expert Systems and an appearance of intelligent DSS;
- an improvement of DSS technology basis.

DSS are becoming a new effective means for problem solution in organizational management systems and everyday business activities. But it is important to emphasize the term "support" in the name of the system. DSS may prompt an unusual question, help to get a deeper insight into the situation, but they do not and will be unable to substitute a creative human being.

References

Alter, S.A. (1980). Decision Support Systems: Current Practice and Continuing Challenges. Reading, Mass., Addison-Wesley Publ. Co.

Anthony, R.N. (1965). Planning and Control Systems: A Framework for Analysis. Boston, Harward University, Graduate School of Business Administration.

Bellew, R.K. (1985). Evolutionary Decision Support Systems: An Architecture Based on Information Structure. In L.B. Methlie and R.H. Sprague, eds., Knowledge Representation for Decision Support Systems. Amsterdam, North-Holland Publ. Co., pp. 147–160.

Bonczek, R.H., Hollsapple, C.W., Whinston, A.B. (1981). Foundation of Decision Support Systems. New York, Academic Press.

Bosman, A. (1982). Decision Support Systems, Problem Processing and Coordination. In M.J. Ginzberg, W. Reitman, E.A. Stohr, eds., Decision Support Systems. Amsterdam, North-Holland Publ. Co., pp. 79–92.

Donovan, J.J. and Madnik, S.E. (1977). Institutional and Ad Hoc Decision Support Systems and Their Effective Use. *Data Base*, 8, No. 3, pp. 79–88.

Ginzberg, M.J. and Stohr, E.A. (1982). Decision Support Systems: Issues and Perspectives. In M.J. Ginzberg, W. Reitman, E.A. Stohr, eds., Decision Support Systems. Amsterdam, North-Holland Publ. Co., pp. 9–31.

Keen, P.G.W. (1980). Decision Support Systems: A Research Perspective In G. Fick and R.H. Sprague, eds., Decision Support Systems: Issues and Challenges. Oxford, Pergamon Press, pp. 23–44.

Larichev, O.I. (1985). Problem of Man-Machine Interaction in Decision Support Systems. In L.B. Methlie and R.H. Sprague, eds., Knowledge Representation for Decision Support Systems. Amsterdam, North-Holland Publ. Co., pp. 27–39.

Larichev, O.I. and Petrovsky, A.B. (1987). Decision Support Systems: State-of-the-Art and Prospects. In: Developments in Science and Technology. Technical Cybernetics. Moscow, VINITI, pp. 131–164, (in Russian).

Sprague, R.H. (1980). A Framework for Research on Decision Support Systems. In G. Fick and R.H. Sprague, eds., Decision Support Systems: Issues and Challenges. Oxford, Pergamon Press, pp. 5–22.

Feedback-Oriented Group Decision Support in a Reference Point Framework

Rudolf Vetschera
University of Vienna, Austria

1 Introduction

Existing approaches to group decision support systems mainly deal with two issues: aggregation of opinions (e.g. Lewandowski et al., 1986; Jarke et al., 1987) and communication among group members (e.g. Turoff/Hiltz, 1982; DeSanctis/Gallupe, 1987) or between members and decision support tools (e.g. Steeb/Johnston, 1981; Huber, 1984).

Group decision making, however, is an iterative process in which aggregation is only part of the entire feedback loop. The other side of the loop, changes that bring opinions closer together, is not supported by current approaches. Their importance, however, is clearly recognized in the literature (Jarke et al., 1987; Lewandowski et al., 1986). This paper is based on a general concept of supporting such feedback processes where individual evaluations are changed so that they more closely correspond to the group evaluation.

The implementation of such changes depends on the individual decision technique. In this paper, we assume that group members use the reference point method (Wierzbicki 1984). This concept is explicitly designed to allow for learning and other phenomena that might cause opinions to change over time and are important for feedback-oriented group decision support. Section 2 introduces the general concept of feedback-oriented group decision support. In section 3, we present a specific model for the reference point method. Section 4 summarizes our results and identifies some topics for further research.

2 General Framework

We consider a decision situation in which a group of M members m has to decide about N alternatives n known to all group members. Each member evaluates the alternatives according to K_m (individually different) criteria k. All objectives are to be maximized.

The choice problem to be solved by the group consists in ranking the c best out of N alternatives. This is a general formulation: For $c = 1$, it is equivalent to the selection of the best alternative, for $c = K$ to the ranking of all alternatives.

In feedback-oriented group decision support, two information flows and processing steps are of importance: information on individual opinions is transmitted to the group level and aggregated. The resulting group opinion is then transmitted back to the members, who use this information to modify their opinions for the next iteration. This paper concentrates on the modification of individual opinions through the group opinion, which is represented by a cardinal evaluation of alternatives.

The group evaluation can be incorporated into the individual evaluations in two ways:

- **Explicitly:** The group evaluation is considered as an additional attribute in the individual evaluations.
- **Implicitly:** The individual evaluation systems are changed to agree with the group evaluation.

The modified evaluation process should fulfill some requirements. We will use the notation $A \succ_m B$ to indicate that group member m initially prefers alternative A to alternative B. Similarly, $A \succ_g B$ represents the group ranking and $A \succ B$ the combined ranking.

Our first requirement concerns **consistency**: If both the group and the member prefer A to B, the combined ranking should also do so. Formally:

$$A \succ_g B,\ A \succ_m B \Rightarrow A \succ B \tag{1}$$

The second requirement concerns the **resolution of conflicts** between individual and group rankings. If the two rankings do not agree, the aggregation process should be controllable by a parameter p. Interpreted differently, this parameter can serve as an indicator showing how much importance must be given to the group ranking in order to achieve a certain combined ranking.

$$\forall A, B:\ B \succ_m A, A \succ_g B\ \exists \bar{p}:\ \begin{array}{ll} p > \bar{p} & \Rightarrow A \succ B \\ p < \bar{p} & \Rightarrow B \succ A \end{array} \tag{2}$$

3 A Reference Point Change Model

In the reference point method, (individual) evaluations of alternatives are described by a reference point and a *scalarizing function s*. In this paper, we will use the scalarizing function $s(w)$ employed in the DIDAS software package (e.g. Wierzbicki, 1984, p.478; Grauer et al., 1984, p.26):

$$s(w_n) = -\min\left\{\min_k \rho w_{nk}, \sum_k w_{nk}\right\} \tag{3}$$

omitting a possible correction term required for computational reasons. In (3), $w_{nk} = q_{nk} - \bar{q}_k$ is the distance of the alternative under consideration (n), evaluated in attribute k as q_{nk}, to the reference point $\bar{q}$. An alternative is considered better the lower its value of $s(w_n)$. The choice of a scalarizing function is not of great importance for our problem, similar models can easily be developed for different functions.

To provide explicit feedback, cardinal values are used for the group evaluation. In this paper, we are not concerned with the method by which individual evaluations are aggregated (see e.g. Lewandowski et al. 1986; Fortuna/Krus 1984). Without loss of generality, we assume that a higher value in this evaluation implies preference by the group.

By treating the group evaluation (attribute g) like all the other attributes, we obtain an extended scalarizing function as

$$t(w, w_g) = -\min\left\{\min_k \rho w_k, \sum_k w_k + w_g, \rho w_g\right\} \tag{4}$$

Implicit modification is obtained by changing the group member's initial reference levels $\bar{q}_k$ to modified levels $\bar{\bar{q}}_k$.

Without loss of generality, we assume that the alternatives are labelled according to the group's ranking, i.e. alternative 1 is preferred to alternative 2 etc. The goal of the group is to find a common ranking of the best c alternatives:

$$\begin{array}{rcll} t(w'_n, w_{ng}) & < & t(w'_{n+1}, w_{n+1g}) & n = 1, \ldots, c-1 \\ t(w'_c, w_{cg}) & < & t(w'_n, w_{ng}) & n = c+1, \ldots, N \end{array} \tag{5}$$

where $w'_{nk} = q_{nk} - \bar{\bar{q}}_k$. These conditions can be reformulated as a set of constraints in a mixed integer optimization problem. Substituting into the scalarizing function and replacing, for ease of exposition, $-min$ by $max-$, yields for two alternatives n and $n+1$:

$$\begin{array}{l} \max\left\{\max\limits_k -\rho(q_{nk} - \bar{\bar{q}}_k), -\sum\limits_k(q_{nk} - \bar{\bar{q}}_k) - (q_{ng} - \bar{q}_g), -\rho(q_{ng} - \bar{q}_g)\right\} < \\ \quad \max\left\{\max_k -\rho(q_{n+1,k} - \bar{\bar{q}}_k), -\sum_k(q_{n+1,k} - \bar{\bar{q}}_k) - (q_{n+1,g} - \bar{q}_g), -\rho(q_{n+1g} - \bar{q}_g)\right\} \end{array} \tag{6}$$

This condition is fulfilled if we can find any z_n so that

$$\begin{array}{rcll} z_n & \geq & -\rho(q_{nk} - \bar{\bar{q}}_k) & k = 1, \ldots, K \\ z_n & \geq & -\sum_k(q_{nk} - \bar{\bar{q}}_k) - (q_{ng} - \bar{q}_g) & \\ z_n & \geq & -\rho(q_{ng} - \bar{\bar{q}}_g) & \end{array} \tag{7}$$

and

$$\begin{array}{rcll} z_n & < & -\rho(q_{n+1,k} - \bar{\bar{q}}_k) + (1 - \lambda_{nk})M & k = 1, \ldots, K \\ z_n & < & -\sum_k(q_{n+1,k} - \bar{\bar{q}}_k) - (q_{n+1,g} - \bar{q}_g) + (1 - \lambda_{nS})M & \\ z_n & < & -\rho(q_{n+1,g} - \bar{\bar{q}}_g) + (1 - \lambda_{ng})M & \\ 1 & \leq & \sum_k \lambda_{nk} + \lambda_{nS} + \lambda_{ng} & \end{array} \tag{8}$$

where $\lambda_{nk}, \lambda_{nS}, \lambda_{ng} \in \{0, 1\}$ and M is a suitably large constant.

Under these conditions, we have to find the most acceptable, i.e. minimal change of the member's evaluation system. The penalty scalarizing function itself cannot be used to evaluate the amount of change because it is intended to represent the member's opinion about the decision problem, but the main purpose of the feedback process is to *change* this opinion.

Implicit change can be measured through deviation variables $\delta_k^+ \geq 0$ and $\delta_k^- \geq 0$ defined by:

$$\bar{\bar{q}}_k = \bar{q}_k + \delta_k^+ - \delta_k^- \tag{9}$$

An objective function can be formulated using these change variables and different norms. Using the ℓ_1 norm, we obtain a linear criterion for implicit change as $\sum_k \delta_k^+ + \sum_k \delta_k^-$.

Explicit change can be measured through a reference level $\bar{q}_g$ for the group attribute, which satisfies conditions (1) and (2) if parameter ρ is chosen large enough so that no alternative is evaluated according to the min-max rule if it exceeds the reference level in all criteria. Combining the two forms of modification, we obtain the following model:

$$\begin{array}{rcll}
\text{minimize} & & \sum_k \delta_k^+ + \sum_k \delta_k^- & \\
\text{minimize} & & \bar{q}_g & \\
\text{s.t.} & & & \\
\bar{\bar{q}}_k & = & \bar{q}_k + \delta_k^+ - \delta_k^- & \forall k \\
 & & & \\
z_n & \geq & -\rho(q_{nk} - \bar{\bar{q}}_k) & n = 1, \ldots, c; \forall k \\
z_n & \geq & -\sum_k (q_{nk} - \bar{\bar{q}}_k) - (q_{ng} - \bar{q}_g) & n = 1, \ldots, c \\
z_n & \geq & -\rho(q_{ng} - \bar{q}_g) & n = 1, \ldots, c \\
 & & & \\
z_n & < & -\rho(q_{n+1,k} - \bar{\bar{q}}_k) + (1 - \lambda_{nk})M & n = 1, \ldots, c-1; \forall k \\
z_n & < & -\sum_k (q_{n+1,k} - \bar{\bar{q}}_k) - (q_{n+1,g} - \bar{q}_g) + (1 - \lambda_{nS})M & n = 1, \ldots, c-1 \\
z_n & < & -\rho(q_{n+1,g} - \bar{q}_g) + (1 - \lambda_{ng})M & n = 1, \ldots, c-1 \\
 & & & \\
1 & \leq & \sum_k \lambda_{nk} + \lambda_{nS} + \lambda_{ng} & n = 1, \ldots, c-1 \\
 & & & \\
z_c & < & -\rho(q_{n,k} - \bar{\bar{q}}_k) + (1 - \lambda_{nk})M & n = c+1, \ldots, N; \forall k \\
z_c & < & -\sum_k (q_{n,k} - \bar{\bar{q}}_k) - (q_{n,g} - \bar{q}_g) + (1 - \lambda_{nS})M & n = c+1, \ldots, N \\
z_c & < & -\rho(q_{n,g} - \bar{q}_g) + (1 - \lambda_{ng})M & n = c+1, \ldots, N \\
 & & & \\
1 & \leq & \sum_k \lambda_{nk} + \lambda_{nS} + \lambda_{ng} & n = c+1, \ldots, N
\end{array} \tag{10}$$

This model itself represents a multi-objective optimization problem, where a trade-off must be determined between explicit and implicit modification. The two extreme solutions can easily be determined by minimizing the two objectives individually. However, it is not possible to obtain the efficient frontier between these two points by minimizing a weighted sum of the objectives because the problem contains integer variables and the set of feasible solutions need not be convex.

It can be shown that, in general, (4) may violate the consistency condition (1). For the optimal modification model considered above, however, this violation does not pose a problem. The constraints of the model ensure that the ranking generated approximates the group ranking, so they preclude possible preference reversal effects.

4 Conclusions and Topics for Further Research

In this paper, we have developed a reference point change model, which allows the consideration of group opinions in individual evaluations using the reference point method. By formally incorporating the group opinion into the individual decision process, we are able to support feedbacks in group decisions, which are of great importance in practical situations.

An important topic for further research concerns the computational efficiency of the process. The model formulated requires the use of $O(N \cdot K)$ zero-one variables. While the solution of a single adaptation problem for one objective can be found in reasonable time, generating trade-off curves showing possible combinations of the two types of modification requires extensive computational effort. Here the special structure of the problem and the close relationship between solutions need to be considered in the solution process.

A further topic relates to practical application. A process that is based on individual group members changing their opinions needs careful validation, preferably through empirical research, to determine the acceptability of the approach to group members.

References

DeSanctis, Gerardine; Gallupe, R. Brent (1987).*A Foundation for the Study of Group Decision Support Systems.* Management Science 33: 589–609.

Fortuna, Zenon; Krus, Lech (1984). *Simulation of an Interactive Method Supporting Collective Decision Making Using a Regional Development Model.* In: Grauer, M.; Wierzbicki, A.P. (Eds.): Interactive Decision Analysis. Proceedings, Laxenburg 1983. Springer, Berlin: 202–209.

Grauer, M.; Lewandowski, A.; Wierzbicki, A.P. (1984). *DIDAS - Theory, Implementation and Experiences.* In: Grauer, M.; Wierzbicki, A.P. (Eds.): Interactive Decision Analysis. Springer, Berlin: 22–30.

Huber, George P. (1984). *Issues in the Design of Group Decision Support Systems.* MIS Quarterly: 195–204.

Jarke, Matthias; Jelassi, M. Tawfik; Shakun, Melvin F. (1987). *MEDIATOR: Towards a Negotiation Support System.* European Journal of Operational Research 31: 314–334.

Lewandowski, Andrzej; Johnson, Sarah; Wierzbicki, Andrzej (1986). *A Prototype Selection Committee Decision Analysis and Support System. SCIDAS: Theoretical Background and Computer Implementation.* IIASA Working Paper WP-86-27. IIASA, Laxenburg.

Steeb, R.; Johnston, S.C. (1981). *A Computer Based Interactive System for Group Decision Making.* IEEE Transactions Systems, Man and Cybernetics 11: 544–552.

Turoff, M.; Hiltz, S.R. (1982). *Computer Support for Group Versus Individual Decisions.* IEEE Transactions Communications 30: 82- 90.

Wierzbicki, A.P. (1984). *The Use of Reference Objectives in Multiobjective Optimization.* In: Fandel, G.; Gal, T. (Eds.): Multiple Criteria Decision Making – Theory and Application. Springer, Berlin: 468–486.

This series reports new developments in mathematical economics, economic theory, econometrics, operations research, and mathematical systems, research and teaching – quickly, informally and at a high level. The type of material considered for publication includes:

1. Preliminary drafts of original papers and monographs
2. Lectures on a new field or presentations of a new angle in a classical field
3. Seminar work-outs
4. Reports of meetings, provided they are
   a) of exceptional interest and
   b) devoted to a single topic.

Texts which are out of print but still in demand may also be considered if they fall within these categories.

The timeliness of a manuscript is more important than its form, which may be unfinished or tentative. Thus, in some instances, proofs may be merely outlined and results presented which have been or will later be published elsewhere. If possible, a subject index should be included. Publication of Lecture Notes is intended as a service to the international scientific community, in that a commercial publisher, Springer-Verlag, can offer a wide distribution of documents which would otherwise have a restricted readership. Once published and copyrighted, they can be documented in the scientific literature.

Manuscripts

Manuscripts should be no less than 100 and preferably no more than 500 pages in length. On request, the publisher will supply special paper with the typing area outlined and essentials for the preparation of camera-ready manuscripts. Manuscripts should be sent directly to Springer-Verlag Heidelberg or Springer-Verlag New York.

Springer-Verlag, Heidelberger Platz 3, D-1000 Berlin 33
Springer-Verlag, Tiergartenstraße 17, D-6900 Heidelberg 1
Springer-Verlag, 175 Fifth Avenue, New York, NY 10010/USA
Springer-Verlag, 37-3, Hongo 3-chome, Bunkyo-ku, Tokyo 113, Japan

ISBN 3-540-53432-6
ISBN 0-387-53432-6