JYVÄSKYLÄ STUDIES IN COMPUTING

84

Iryna Yevseyeva

Solving Classification Problems with Multicriteria Decision Aiding Approaches

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 84

Iryna Yevseyeva

Solving Classification Problems with Multicriteria Decision Aiding Approaches

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Villa Ranan Blomstedtin salissa
joulukuun 20. päivänä 2007 kello 14.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in the Building Villa Rana, Blomstedt Hall, on December 20, 2007 at 2 p.m.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2007

Solving Classification Problems with Multicriteria Decision Aiding Approaches

JYVÄSKYLÄ STUDIES IN COMPUTING 84

Iryna Yevseyeva

Solving Classification Problems with Multicriteria Decision Aiding Approaches

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2007

Editors
Tommi Kärkkäinen
Department of Mathematical Information Technology
University of Jyväskylä
Irene Ylönen, Marja-Leena Tynkkynen
Publishing Unit, University Library of Jyväskylä

URN:ISBN:9789513930974
ISBN 978-951-39-3097-4 (PDF)

ISBN 978-951-39-3049-3 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2007

ABSTRACT

Yevseyeva, Iryna
Solving classification problems with multicriteria decision aiding approaches
Jyväskylä: University of Jyväskylä, 2007, 182 p.
(Jyväskylä Studies in Computing
ISSN 1456-5390; 84)
ISBN 978-951-39-3097-4 (PDF), 978-951-39-3049-3 (nid.)
Finnish summary
Diss.

In classification problems with decisions highly dependent on subjective information and preferences of a particular decision maker, the multiple criteria decision aiding approaches can be applied. The decision aiding aspect of these techniques allows providing a significant help for the decision maker by defining the problem, formalizing it, and suggesting supportive methods. In this work, we are motivated by the idea of structuring the field of classification based on multicriteria decision aiding approaches. We present a comprehensive state-of-the-art survey of classification with multicriteria decision aiding approaches. We categorize the existing methods and discuss in more detail the most widely-spread methods. For each method, we study advantageous and disadvantageous properties and look for similarities and peculiarities. In addition, we compare different properties of the existing methods. Such analysis indicates gaps in the existing methods and has motivated us to develop two new methods that would fill some gaps.

In particular, we have extended stochastic multicriteria acceptability analysis for nominal classification and have created the SMAA-Classification method. The method does not need parametrical information to be specified but assumes that the decision maker may provide some assignment example(s) for each class. The output of the method contains information about acceptability of each alternative to be assigned in each class. On the other hand, for the situation where there is no possibility to specify assignment examples for classes, verbal decision analysis methods may be applied for ordinal classification. They assume absence of any other information about classes but their number and order. The methods are interactive. In a dialog regime, the decision maker has to classify some alternatives selected by the method, but not all. We have developed the Dichotomic Classification method in the framework of verbal decision analysis.

We illustrate the methods developed with simple examples and perform some numerical experiments for the data sets available in the multicriteria decision aiding literature. The results of these tests speak for the efficiency of the methods developed in this work. The introduced Dichotomic Classification method is applied to neuropsychological diagnostics, in particular, diagnostics of Attention Deficit - Hyperactivity disorder. We implement the methods introduced and some of the existing methods in the framework of the multicriteria decision support system developed. For completing the up to date picture of the classification with multicriteria decision aiding approaches, we provide some recommendations for selecting a method to be used.

Keywords: multicriteria decision aiding, classification, multicriteria acceptability analysis, verbal decision analysis, neuropsychological diagnostics

Author Iryna Yevseyeva
Department of Mathematical Information Technology
P.O. Box 35 (Agora)
FIN-40014 University of Jyväskylä
Finland
`iyevsev@cc.jyu.fi`

Supervisors Professor Kaisa Miettinen
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Doctor Pekka Salminen
School of Business and Economics
University of Jyväskylä
Finland

Reviewers Doctor Vincent Mousseau
LAMSADE
Université Paris Dauphine
France

Doctor Eugenia Furems
Institute for System Analysis
Russian Academy of Sciences
Russia

Opponent Assistant Professor Luís Miguel Cândido Dias
School of Economics
University of Coimbra
Portugal

ACKNOWLEDGEMENTS

I wish to thank all those who helped me to carry out this research. First of all, I wish to express my deep gratitude to Professor Kaisa Miettinen, who suggested me to expand my research beginnings to the postgraduate studies and encouraged me to pursue doctoral degree. During this work she has been assisting me with her comments, questions, and advices concerning methodological, structural and practical aspects of this work. In addition, she has been motivated me by example of her own work and contributions in the field of multiobjective optimization.

I would like to thank to Doctor Pekka Salminen for discussions, professional guidance and his sharing expertise in multicriteria decision aiding. Many thanks to Licentiate Pekka Räsänen for his cooperation and guidance in neuropsychological diagnostics.

I am especially grateful to external reviewers Doctor Eugenia Furems and Doctor Vincent Mousseau for carefully reading the thesis and for their valuable feedback. Their different viewpoints (supported by the multicriteria decision aiding schools they belong to) helped me to improve the thesis considerably.

I am also thankful to Professor Pekka Neittaanmäki for providing me comments and guidance that speed up the process of the thesis preparation.

My thanks go to Professor Risto Lahdelma, Professor Eduard G. Petrov, Assistant Professor Carlos M. Fonseca and MSc Philippe Nemery de Bellevaux for reading and commenting parts of this manuscript and to Alexander Maslov for providing me some programming hints.

For linguistic proofreading of this manuscript, I would like to thank Professor Kaisa Miettinen and Steve Legrand. For the translation of Finnish summary, thanks to Licentiate Timo Aittokoski and Professor Kaisa Miettinen.

For financial support during my postgraduate research, I wish to acknowledge the University of Jyväskylä (especially, for providing me with the one-year rector grant for starting my research work), the COMAS Graduate School in Computing and Mathematical Sciences at the University of Jyväskylä and the Academy of Finland (grant number 104641 managed by Professor Kaisa Miettinen), as well as to Ella and George Ehrnrooth foundation, Technological foundation, Ellen and Artturi Nyyssonen foundation.

I wish to thank to the EURO Working Group on Multicriteria Decision Aiding and their members for the active discussions during meetings and to the Association of European Operational Research Societies for organizing meetings and summer schools for young researchers in the field of Operational Research.

I would like to express my appreciations to the researchers of the Department of Mathematical Information Technology and Optimization Group at the University of Jyväskylä for their remarks and comments during my work and to the staff of the Faculty of Information Technology at the University of Jyväskylä for assisting in solving organizational issues.

My sincere gratitude goes to my family, to my sisters, Elena and Nataliia, and, especially, to my dear parents, Victor and Lidia, for their unconditional faith and all-ways support immensurable for my well-being. Finally, I am thanking to my friends for bringing joy and spirit into my life and sharing with me being moments.

Jyväskylä, 23 October 2007,

Iryna Yevseyeva

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

The main topic of this work is classification that considers assignment of a set of given alternatives into a set of categories or classes. Classification is necessary in a variety of problem areas such as medicine, financial management and economics as well as marketing research, human resources management, production system management and technical diagnostics, pattern recognition, environment ecology and energy management ([3], [10], [37], [74], [169]). For example, a doctor performs a diagnosis of a patient based on the analysis of his or her symptoms; a financier estimates credit risk before investments and an analyst of a production system controls and monitors complex functioning systems. The existence of such a wide range of applications creates the preconditions for formalizing classification problems and developing tools for solving them efficiently.

Traditionally, classification problems have been solved by parametric techniques of multivariate statistical analysis, such as linear [48] and quadratic [147] discriminant analysis, and econometrics, such as logit-probit analysis [18]. These methods are still in use; however, they have been criticized due to difficulties of practical application when statistical assumptions and restrictions cannot be specified precisely. On the other hand, development of artificial intelligence, in particular machine learning, allows considering non-parametric techniques, such as neural networks, expert systems, and decision trees [39], for classification problems. Other methods for classification can be found in operational research, in particular, in multicriteria decision aiding (e.g., utility theory [159], outranking approach [136], verbal decision analysis [98], [100], rough sets theory [55], and fuzzy sets theory [166]).

The selection of a classification method to be used depends greatly on the availability of initial information, the type of such information, completeness of this information, the level of uncertainty, and the requirements for the final decision. Most of the currently developed methods for classification analyze large data sets and search patterns of similar behavior for parts of this set. However, there are classification tasks in which knowledge and preferences of an expert or a decision maker (DM) play a significant role. For example, the decision may be highly dependent on subjective information and personal thoughts of a DM, his or her beliefs and/or intuition. In such situations, the multiple criteria decision aiding (MCDA) methods can be applied. The decision aiding aspect of these techniques allows providing a significant help for the DM by structuring the problem, formalizing it, and suggesting supportive methods. In this work, we concentrate mainly on the methodological part of classification problems with the MCDA theory.

In MCDA, the DM is a person, who is responsible for making hard and/or complex decisions. There may also be an expert involved who may share his or her expertise in the problem area of the decision to be made. Neither the DM no the expert are supposed to master MCDA techniques; rather, it is the analyst who is supposed to be familiar with MCDA methodologies and techniques and support in decision making. With MCDA methods, the DM is involved in the process of decision making by providing not only information necessary for the method but also his or her own knowledge of the decision area and preferences according to the situations considered. The expert may take part by sharing his or her expertise in the problem domain. In case of some methods, they also participate in the decision making by interacting with the MCDA method at intermediate stages.

Elements involved in the decision aiding could be characterized as knowledge and preferences. Preference can be understood as relating to opinions of the DM about desirability for some features to have some values when compared to other values. In other words, we can say that there is direction for some feature values that make the decision to be more attractive and desirable to the DM. On the other hand, knowledge does not have this kind of direction but is related more to the experience, believes and intuition of the DM or the expert. We refer to both types of information as preferences and do not treat them in any specific way.

By the concept "multicriteria" we refer to the idea that each alternative can be estimated from several usually conflicting points of view; here we deal with so-called criteria measurements. Thus, several conflicting criteria have to be considered simultaneously. The differences in the measurements of criteria as well as possible dependencies between them should be taken into account. The fact that real-world alternatives may be described by a large number of criteria is also important. The problem area of MCDA applications varies from everyday problems, such as "what to buy for dinner" and "where to go on the weekends", to vitally important decisions, such as "selection of the profession" or "choice of country for living".

There are two different approaches for representation and, consequently, solving multiple criteria problems [108]: multiobjective optimization and multiattribute decision analysis. The latter one is also called multicriteria decision aiding [140], and that is how we refer to it in the present work. The difference between the two approaches is in treating the feasible set of alternatives. Indeed, in multiobjective optimization the set of alternatives is not available before the actual optimization procedure starts. It is assumed that there is an infinite number of alternatives to be evaluated and that they are represented by decision variables and restricted by constraints in the decision variables space. The multiobjective optimization problems are also called continuous. By contrast, in multicriteria decision aiding, the set of alternatives to be estimated is discrete, finite, and usually available in advance and/or predefined by the DM [108]. For comprehensive surveys on multiobjective optimization, we refer to [40], [41], [108].

There is a large amount of literature devoted to MCDA problems. Surveys of this field may be found in [16], [19], [22], [44], [52], [140], [158]. A large bibliography for MCDA is presented at the web site of LAMSADE laboratory [91]. A lot of new ideas are presented twice per year at the conferences of the EURO Working Group on MCDA [43], as well as on the international conferences of the Multiple Criteria Decision Making (MCDM) society [67] and the Association of European Operational Research Societies (EURO) [4] that take place biannually, to name a few.

In this work, we consider classification of a set of discrete alternatives. That is why we concentrate on solving it with MCDA methods. Traditional problems to be solved by MCDA methods are the choice of the best alternative (or several best ones) and ranking of the set of given alternatives. Examples of them are the selection of a car to buy or the university where to study; or ranking of participants in a competition. The approach of solving classification problems by means of MCDA is quite modern. The first MCDA method for classification was developed in the late 70s - early 80s by Moscarola and Roy [111], [136] and was limited for three classes only. Later, this method was developed for an arbitrary number of classes [165]. The development of other classification methods within the MCDA framework was undertaken during the 80s - 90s, e.g., in [9], [37], [56], [72], [97], [100], [105], [131]. Recently, a number of Masters and PhD Theses have been written about this subject [26], [110], [121], [157], to name a few. Finally, the book on MCDA classification methods by Doumpos and Zopunidis [37] and the special issue about classification with MCDA of the European Journal of Operational Research [168] appeared in 2002.

In this work, we are motivated by the idea of structuring the field of classification with MCDA approaches and we follow two main goals. First, we want to perform a representative review of the state-of-the-art of MCDA methods for classification and provide a more detailed discussion of the most developed and widely-spread methods. Second, we pursue the development of new MCDA-based methods for classification that would improve the performance of the existing methods and extend the MCDA methodology for classification problems.

At the beginning of this work, we introduce basic terms and concepts used in the MCDA methodology. We also describe decision aiding approaches and the decision aiding process itself. Then, we define a classification problem and the types of classification. We present an extensive survey of the existing most widely-used methods and discuss their advantageous and disadvantageous properties as well as describe their extensions and applications. We summarize the discussion of each method with some concluding remarks.

Besides some exceptions like [26], [37], [161], [172], there are very few reviews of different approaches to classification within the MCDA framework available in the literature. In this work, we try to fill this gap to some extent. We also compare the features of the surveyed methods in order to define their advantages and disadvantages as well as their similarities and peculiarities. The comparison of the existing methods motivates us to improve them and create some new ones. It also allows us to provide some recommendations for the selection of a method to be used on each particular situation. There is no single best method to be used for every case. That is why it is important to provide the DM and/or analyst with some tips for the selection of an appropriate method depending on the available initial information, requirements to the results of classification, the decision aiding process itself, and the nature of the classification problem.

As mentioned before, there are differences between the classical approach (e.g., statistics and artificial intelligence) and MCDA. Indeed, the latter involves the DM in the classification problem. In addition, MCDA usually assumes that the alternatives to be classified are estimated on a set of conflicting criteria, each of which provides not only a description of some property (like attributes in classical approach) but includes additional preferential information (e.g., bigger values are assumed to be preferred or more desirable to smaller ones). Another aspect considered in MCDA is the order of the classes. In classical approaches, the classes are described in a nominal way and the order is not considered. Even though the main focus of the methods that stem from the classical approaches (that is the accuracy of the results obtained from the model) has a high importance, taking into account the issues considered by MCDA may improve and/or simplify the performance of the classification models.

On the other hand, deep involvement of the DM into the decision aiding process brings new challenges for the development of robust models that represent informal thinking of the DM as closely as possible to the real-life situation. Such challenges include (but are not limited to) taking into account cognitive sides of the DM when expressing his or her preferences, representing them in the form understandable for the DM, and providing control to the DM during the process of decision aiding.

The cognitive abilities (e.g., short-term memory) of the DM are limited [144]. People cannot deal with large volumes of information simultaneously and in order to minimize efforts they tend to simplify their strategies and operations. Moreover, humans are not accurate and may make mistakes as well as be inconsistent with their preferences. However, these facts are not usually taken into account when developing MCDA methods.

In some cases, the DM may not be fully aware of his or her own preferences, in particular, if they have to be defined in terms and parameters of some MCDA model. The DM may not be able to understand the exact meaning of such terms and to provide the required information. However, he or she can give some decisions that he or she used to make in the past or is ready to think about the consequences of subset of possible decisions. The survey of the existing methods shows that Stochastic Multicriteria Acceptability Analysis (SMAA) [83] provides such possibilities for preference modeling for choice and ranking types of MCDA problems; however, there is some lack of methods for solving classification problems. That is why we have developed a so-called SMAA-Classification method to be used in some types of classification problems.

The problem may be described not only with numerical data but using natural language, that is, verbal data. The latter may be the only way to express the complexity of the situation. Thus, the DM expects to have a possibility to express his or her preferences as well as to receive the results of the inference in natural language. However, most of the MCDA methods operate only with numerical values and, moreover, some of them assume the DM to be able to transform the qualitative preferences to the quantitative ones. Such a transformation may be misleading and cause error propagation. By taking into account the ability to treat verbal values it is possible to avoid such weaknesses. In this work, we search for methods that consider cognitive sides of the DM and allow operating with verbal estimations expressed in natural language. The methods that satisfy such features have been developed in the framework of verbal decision analysis [100]. We wish to improve the existing verbal decision analysis methods for classification problems and introduce a method called Dichotomic Classification in the verbal decision analysis framework.

The two new MCDA methods presented in this work are very different and are used for different types of classification problems with different assumptions regarding initial and resulting data as well as the process of decision aiding itself. Indeed, SMAA-Classification does not use the information about the order on the set of classes assuming them to be nominal, and it operates with numerical values on the scales of criteria and/or attributes. The method assumes the availability of a set of examples of the classifications, so-called assignment examples, the decisions that the DM used to make in the past; or decisions made for realistic alternatives not considered in the set of given alternatives; or DM's readiness to make decisions for a limited set of alternatives from the set of given alternatives [72]. Based on the set of assignment examples, the method does a classification and searches in an inverse manner for the preferences that support such a classification. The method can be applied in an interactive way and provides the DM with an opportunity to learn how his or her own decisions influence the MCDA classification model developed. On the other hand, the Dichotomic Classification method is an interactive method that assumes an order between the classes, and an ordinal scale of verbal values for each criterion. It supposes that there is no initial information about the preferences of the DM available before the decision aiding process starts and elicits preferential information iteratively.

We demonstrate the methods by some illustrative examples and perform numerical experiments with data sets available in the MCDA literature. We apply the Dichotomic Classification method to the neuropsychological diagnostics of Attention-Deficit Hyperactivity Disorder (ADHD). The developed and some studied methods are implemented within a multicriteria decision support system that is equipped with a comfortable interface allowing user-friendly interaction between the DM (and/or analyst) and the system.

To summarize, following our main motivation to structure the field of classification with MCDA approaches, we define the next goals of this work. We wish to survey the existing MCDA methods for classification and present a comprehensive state-of-the-art of classification with MCDA approaches. Some categorization of the existing methods should be provided and the most widely-spread methods should be discussed in more detail. Their advantageous and disadvantageous properties should be defined. The comparison of existing methods should indicate the gaps and motivate us to create some new methods that would fill some gaps. The methods developed should be compared to the existing methods and their performance should be tested on the same data sets. We also would like to provide some recommendations for selecting an MCDA method for classification among the discussed methods. The methods developed and some ones studied should be implemented in a multicriteria decision support system. The method(s) developed should be applied to the neuropsychological diagnostics. Finally, the conclusions of this work should be provided and directions for future research outlined.

The rest of the work is organized as follows. In Chapter 2, we present the main concepts and definitions of MCDA that will be used throughout this work, we also formulate the MCDA process and discuss different types of classification problems. We survey and categorize the MCDA methods for classification in Chapter 3; some preference elicitation procedures are presented there as well. In Chapter 4, we introduce some new methods developed for different types of classification. In particular, a new SMAA-Classification method and a new Dichotomic Classification method are proposed. They are illustrated with simple examples. Then, in Chapter 5, we compare methods surveyed and developed according to the information used, the results obtained and the decision aiding process itself. We provide the guidelines for selecting a method to be used for the classification with MCDA approaches. Then, the results of the numerical experiments made for the developed and some studied methods are presented in Chapter 6. We also describe the application of the Dichotomic Classification method to the neuropsychological diagnostics of Attention-Deficit Hyperactivity Disorder and a developed multicriteria decision support system in Chapter 7. Finally, we draw some conclusions of the work presented and discuss some directions for further research in Chapter 8.

2 MCDA PRELIMINARIES

In this chapter, we consider some basic concepts of the MCDA theory, actors that take part in the decision aiding, and the decision aiding process itself. We also have a brief look at the different decision aiding approaches, theories and problem types. Finally, we introduce types of classification problems and formulate classification model in MCDA terms. As we have already mentioned in the introduction to this work, MCDA problems involve conflicting criteria. So, to win in one of them, we have to lose in another. Here, we also introduce the types of MCDA problems in which the solution can be aided. Particular attention is paid to the classification type of MCDA problem on which this work is concentrated. This chapter is mostly based on the material presented in [100], [140], [156], [158].

We begin by defining the concepts and aims of decision aiding. By decision aiding, we mean assisting in the decision making when there are several conflicting criteria to be considered simultaneously. It is based on the development of a reality model by questioning the person who is assisted in the decision making. According to Roy [140], "decision aiding is the activity of the person who, through the use of explicit, but not necessary completely formalized models, helps obtain elements of responses to the questions posed by a stakeholder of a decision and usually towards recommending, or simply favoring, a behavior that will increase the consistency between the evolution of the process and the stakeholder's objectives and value system". Thus, the idea of the decision aiding is to assist the DM in structuring the problem, acquiring his or her knowledge and preferences about the problem area and understanding his or her own preferences and, finally, proposing a decision or decisions that best match the DM's long-term goals and possible environmental conditions.

2.1 Basic Concepts

In MCDA theory, some terms and concepts may be defined in a slightly different way and under different titles. In this section, we specify the concepts used throughout this work, such as model, actors, alternatives, criteria and attributes, and relations. We also consider the types of problems in which the MCDA theory may assist.

2.1.1 Model

The model is a fragment of reality represented schematically and formally by an observer (analyst and/or DM, see Section 2.1.2), in order to help to facilitate the decision aiding process. In what follows, we refer to such a model as a reality model. A model can be formal (e.g., defined by mathematical expressions) or judgemental (e.g., defined by the set of decision rules). A formal model has a structure (e.g., defined in the form of equations) and parameters (e.g., in the form of variables in equations). Determination of the structure and parameters for a formal model or the set of rules for a judgemental model is called model identification [132].

One may ask: "How objective is the model?" There is no absolutely "true" or "false" model because it is always only a fragment of reality. The usual way to consider the goodness of a model is its ability to work efficiently for a given problem. That is why limiting the scope of the model is considered to be an art [140]. Every model is limited by the goals that it should serve. Identification of a model assumes transformation from informal and ambiguous information obtained from the expert and/or DM to a more formal reality model that still contains ambiguity, but it is represented in an unambiguous way [156].

The development of the model is very important. As it is often quoted, "a problem well-stated is a problem half-solved" (John Dewey, philosopher). In order to proceed with decision aiding, it is necessary to elicit preferences of a DM and, based on them, to construct a formalized reality model. Let us point out that general structuring of the problem is not the main emphasis of this work. We concentrate on the methodological part of the classification problem and assume the initial information (not necessary precise and well-defined) to be given.

2.1.2 Actors

Persons that are taking part in the decision aiding process are called actors in terms of MCDA. At least two persons are typically involved in the process of decision aiding: DM and analyst.

A DM is a person who is being aided and who takes the responsibility for accepting the final decision. The DM is not supposed to know the MCDA methods, moreover, sometimes he or she is not an expert in some part of the large problem area where the decision is made. However, his responsibility for the final decision forces him to understand the main structure of the problem. Usually, he or she should be able to define the main elements needed for the model, to express ideas, knowledge and preferences. But this does not mean that ideas, knowledge and preferences of the other participants (for instance, experts) are not taken into account.

An analyst is a person who helps the DM to investigate and to structure the problem, to acquire expert's and/or DM's knowledge, and to understand expert's and/or DM's preferences. The analyst creates a model of the problem and chooses a method for obtaining a solution. He or she proposes a decision or decisions that best match the DM's long-term goals and possible environmental conditions. The analyst is usually a specialist (system analyst, system designer, economist, statistician) who is familiar with the MCDA techniques and methods. He or she should preferably be closely acquainted with the problem area of application and clearly understand the definitions and dependencies between the elements in order to limit the scope of the model; he or she should be able to present a fragment of reality (as closely as possible) and to select a proper method for aiding.

Usually, some other people, such as experts, take part in the process of decision aiding. They are professionals in the problem area or some part of it. They influence the decision aiding process by expressing their knowledge or by helping in the construction of the model. However, usually, they are not responsible for the final solution. On the other hand, it may be possible that the DM understands too little some parts of the problem area. Then, the task of the expert is to provide all his or her professional expertise for the DM to support his or her final decision. For instance, we may consider the situation where the boss of some company (who is the DM) has to listen to several managers (who are experts in human resources, financial management and public relations) and make the final decision taking into account their opinions. However, in what follows, the DM and the expert are regarded as the same person (and called the DM).

On the other hand, there are situations where several DMs are taking part in the decision process. In this work, most of the methods are designed for a single DM; however, some allow the presence of several DMs. In these cases, we must consider the aggregation of preferences of multiple DMs.

2.1.3 Alternatives

A set of given alternatives, denoted as X, is a set of actions or objects each of which should be estimated directly (by the DM) or indirectly (by some method) for the solution of the problem considered (see Section 2.1.6). Here, the terms alternatives, actions and objects are treated as synonyms. An alternative $x_i \in X$ is a vector of criterion values (or a tuple of attribute values). (The definitions of criterion and attribute are given below.) Usually the set of given alternatives is available or defined by the DM. (Otherwise, it can be defined as a Cartesian product of scales of criteria values.)

The set of alternatives can be:

- *finite* if it is possible to define all members of the set of alternatives or *infinite* if it is not possible;
- *stable* if the number and content of alternatives from the set of given alternatives cannot be changed during the decision aiding process or *evolutive* if

the set of alternatives changes during the decision aiding process;

- *comprehensive* if accepting one alternative as a final decision excludes the possibility of accepting any other one or *fragmented* if a combination of alternatives can be the final decision.

According to Roy [140], a *potential alternative* is a *realistic* or *fictitious* alternative temporary judged as being realistic by at least one actor. The potential alternative is included in the decision aiding process. This alternative is *real* or *realistic* if it can be a final decision or *fictitious* if it can assist in the decision aiding process but does not exist in real life.

2.1.4 Criteria and Attributes

In MCDA, the alternatives are estimated on a set of criteria and/or attributes denoted as G. The set of criteria and/or attributes defines the main features of each alternative from the set of given alternatives. Each criterion or attribute $g_j \in G$ defines one feature of the alternative. The criterion g_j taking its values from the ordered scale of values $S(g_j)$. Each criterion has to be either maximized or minimized. On the other hand, the attribute may have nominal or ordinal scale; however, it is neither maximized nor minimized.

In what follows, we assume without any loss of generality that preferences increase with the increased value on each criterion. In other words, the maximal value on the criterion is the most attractive or desirable for a DM. Thus, we consider a maximization problem as a subject of intentions of DM in the framework of the decision aiding process.

According to Vincke [158], a criterion is: "a function g_j, defined on X, taking its values in a totally ordered set, and representing the DM's preferences according to some point of view". Criteria may have verbal or numerical values on their scales. Roy in [140] defines criterion with numerical values on the scale as "a mapping g_j from a set of alternatives X to a numerical scale, such that it appears meaningful to compare two alternatives x_i and x_r according to a particular point of view, on the sole basis of their evaluations $g_j(x_i)$ and $g_j(x_r)$. This implies that a criterion induces a preference relation on the set X".

In MCDA, the set of criteria can be used if it has following properties [140]:

- *completeness* if all criteria that allow distinction of alternatives from the set of given alternatives were taken into account. That means that there is no pair of alternatives, for which it is possible to say: "x_i is preferred to x_r, and x_r is preferred to x_i" and for which the following preference relations (see Section 2.1.5) are true: $x_i P x_r$ and $x_r P x_i$. If this property is satisfied, some important criteria are not taken into account, and we have an *incomplete* set of criteria;

- *cohesiveness* if two alternatives are indifferent (have the same values on all criteria), then improving value on one criterion of one alternative x_i and deteriorating value on some other criterion of some other alternative x_r would reflect the following DM's preferences "x_i is preferred to x_r" (see Section 2.1.5): $x_i P x_r$.

- *non-redundancy* if each criterion from the set plays a significant role and removing at least one criterion leads to a violation of one of the two properties considered above.

The set of criteria that satisfies these three properties is called coherent [140].

On the other hand, in the classical approaches to classification (e.g., statistics and artificial intelligence), the alternatives to be classified are estimated with attributes that are neither maximized nor minimized. According to Roy [140], the attribute defines "a simple characteristic or sign that serves as a basis of an estimation or assessment whose only purpose is to discern or distinguish", when compared to a criterion that serves as "a basis for preferential judgement". The attributes may have numerical or verbal values on their scales as well as have ordered values that, however, cannot be used for the preferential judgement but only distinguishing judgement.

For some types of classification problems, description of alternatives by attributes is much more natural (like nominal classification, see Section 2.3.1). In an ideal case, the method should be able to consider both criteria and attributes (for instance, in the way it is done in the rough sets approach, see Section 3.7.2).

2.1.5 Relations

The comparison of alternatives on a set of criteria represents one of the most important questions in the decision aiding theory. In order to model the DM's preferences, it is necessary to define relations between pairs of alternatives. There are four basic binary relations, strict preference, weak preference, indifference, and incomparability, for comparing any two potential alternatives [140].

The *indifference relation* between two alternatives x_i and x_r, denoted as $x_i I x_r$, "corresponds to the existence of clear and positive reasons that justify equivalence between these two alternatives" [140]. Such a situation appears when two alternatives x_i and x_r are equally preferable, in other words, they are of equal importance to the DM. The indifference relation is reflexive and symmetric. The reflexivity property assumes that $x_i I x_i$ is true and symmetry property assumes that the order of alternatives in the notations $x_i I x_r$ and $x_r I x_i$ makes no difference.

The *strict preference relation* of one alternative x_i over the other one x_r, denoted as $x_i P x_r$, "corresponds to the existence of clear and positive reasons that justify significant preference in favor of one (identified) of the two alternatives" [140]. Such a situation appears when one alternative x_i is better than the other alternative x_r for the DM. The strict preference relation is asymmetric and nonreflexive. The asymmetry property assumes that the order in which alternatives appear in relation makes a difference and nonreflexivity relation assumes that $x_i P x_i$ is false.

We should note here that the strict preference relation is very similar to the so-called *dominance relation* that is a concept used in multicriteria optimization for denoting a similar situation, where the alternative x_i should be better or equal to the other alternative x_r on all criteria (to be maximized):

$$x_i \text{ D } x_r \ (x_i \text{ dominates } x_r),$$
$$\text{if } g_j(x_i) \geq g_j(x_r) \text{ and } g_f(x_i) > g_f(x_r) \text{ on at least one criterion } g_f,$$
$$f, j = 1, \ldots, n; j \neq f; i, r = 1, \ldots, m; i \neq r.$$

Such a relation is asymmetric and transitive [158]. However, such a dominance relation does not include preferences of the DM. In MCDA, there are other definitions of dominance relation that include preferences of the DM (see, for instance, [57] and [100]).

The *weak preference relation*, denoted as $x_i Q x_r$, "corresponds to the existence of clear and positive reasons that invalidate strict preference in favor of one (identified) of the two alternatives but that are insufficient to deduce either strict preference in favor of the other alternative or indifference between two alternatives, thereby not allowing either of the two preceding situation to be distinguished as appropriate" [140]. Such a situation may appear when the DM hesitates to make a precise judgement about preference or indifference between two alternatives x_i and x_r. The weak preference relation may be defined with different thresholds (for instance, see Section 3.1.1). The weak preference relation is asymmetric and nonreflexive.

The *incomparability relation*, denoted as $x_i J x_r$, "corresponds to an absence of clear and positive reasons that justify any of the preceding relations" [140]. This defines the situation where the alternative x_i is not in any of the above-mentioned relations with the alternative x_r, or the alternative x_r is not in any of the above-mentioned relations with the alternative with x_i. The incomparability relation is symmetric and nonreflexive.

Usually, in order to model preferences of a DM, a combination of several binary relations is considered and is called system of preference relations. For developing different systems of preference relations, more complex relations that combine several basic ones have been developed, for instance, outranking relation (for details, see [140]).

The *outranking relation*, denoted as $x_i S x_r$, defines the situation in which the preference (strict or weak) or indifference relation is true, that is $x_i P x_r$ or $x_i Q x_r$ or $x_i I x_r$. For more information on other preference relations and their properties, we refer to [140] and [158].

As we have mentioned earlier, we do not specifically make a distinction in the information considered in decision aiding between knowledge and preferences; however, there are different possibilities for such division. For instance, from a classical point of view (e.g., artificial intelligence), in particular from expert systems, we know that knowledge can be understood as all information acquired from an expert for constructing decision rules. Thereby, when adapted to MCDA, knowledge can be referred to as all kinds of information (e.g., initial information, or information obtained during the decision aiding process) that does not require the DM to make pairwise comparisons of elements (e.g., alternatives, criteria, criteria values, and classes) and to say which of two the elements is better. On the other hand, such information that assumes explicit ordering of elements by the expert and/or DM can be referred to as preferences. (Implicit ordering that can be obtained from knowledge, e.g. order of the degrees of air temperature, can be exclude from preferential information.)

However, there is another opinion about this issue. In [50], all information obtained from an expert is considered as knowledge, while information received from the DM is treated as preferences, thus, the expert may express only knowledge, and the DM only preferences. Then, relations can be defined for knowledge kind of information (for details, see [50]). We do not introduce such relations here because we do not use them. And, as mentioned earlier, we do not divide the information provided by the DM and/or the expert in any specific way on knowledge and preferences, and we refer to both types of information as preferences and do not treat them in any specific way. Next, we consider the types of MCDA problems that the DM may meet.

2.1.6 MCDA Problem Types

In MCDA theory, it is assumed that when making a decision the DM usually follows one or several of the following goals: to define the structure of the problem in terms of the MCDA (to specify the set of given alternatives, criteria, etc.), to select the best alternative from the set of given alternatives, to rank the set of alternatives, to classify the set of alternatives into a predefined set of classes. According to these goals, the MCDA problem types are defined. In some literature, for example, in [140], the problem types are called recommendations. Next, we describe the possible types of MCDA problems.

Description of the problem: it is necessary to define the set of alternatives, the set of criteria and/or attributes on which these alternatives will be estimated, and create the model of the problem. This step is included in each type of MCDA problem; however, it is treated separately here because it can be the only goal of decision aiding.

Choice of the best alternative: during the process of decision aiding the DM should find the smallest number of the best possible alternatives. For example, it may be necessary to choose only one university for studying, or to choose a particular person for the position of a manager.

Ranking the set of alternatives: it is necessary to order the alternatives from the given set according to the DM's preferences. An example of such a problem is allocating the participants of a competition between the best and the worst.

Classification of alternatives: alternatives should be assigned into one or several classes from the set of possible ones. Such kind of problem appears, for instance, in diagnostic tasks (e.g., medical, technical, and financial). For example, a doctor defines a disease according to the patient's symptoms. Another example is from everyday life: we may sort "to-do" tasks for the next day into four classes, such as "must do", "wish to do", and "can do".

There are other possible combinations of the types of MCDA problems mentioned. (For examples, see [102].) For instance, choice and ranking can be combined when it is necessary to select several best alternatives. Then, the example mentioned for a choice type of problem can be altered to get the alternative form: it is necessary to choose two persons for the positions of secretaries. In next section, we consider the stages of analysis that the DM together with the analyst have to go through for solving a problem of any type.

2.2 Decision Aiding Process

The process of decision aiding consists of several issues to be considered. At the beginning, the analyst together with the DM should be engaged in structuring the problem area. At this so-called *initialization step*, it is necessary to define the goals that should be achieved as a result of decision aiding. With respect to them the required initial information is specified. This assumes that a set of alternatives (see Section 2.1.3) is defined. The set of criteria (see Section 2.1.4), on which the alternatives are estimated is also selected. Here, the requirements for the set of criteria are checked (see Section 2.1.4). Then, the type of the problem situation (see Section 2.1.6) is established, as well as the consequences of possible decisions according to the goals are investigated. The additional information (e.g., set of classes for the classification problem and order of classes for the ordinal type of classification) as well as the requirements for the resulting information are also defined at this step.

Next, at the *method selection step* the MCDA method (that may follow one of the MCDA theories presented in Section 2.2.3) is selected from the set of possible ones with respect to the structure defined at the previous step and according to the type of the problem (see Section 2.1.6). In this work, we give some recommendations for the selection of the MCDA method for the classification type of problem (see Section 5.2).

In order to develop a model, some methods use parameters, such as criteria weights (see Section 5.1.9) and/or thresholds (see, for example, Section 3.1.1), and other methods need decision rules (see, for instance, Sections 3.5.1 and 3.7). These are usually defined by the DM. However, this task may be difficult and cognitively loading for the DM. That is why the special procedures that may simplify the preference elicitation (e.g., see Section 3.2) or may assist in rules inferring (e.g., see Section 3.5.1) have been developed. Thus, if necessary, at the *preference elicitation step*, the parameters of the model or decision rules are defined by the DM or may be extracted with respect to some procedure.

During the decision aiding process, the analyst and/or the DM may face a number of problems. These are related to uncertainties that appear when simplifying the real-world situation to formal models and may be caused by different factors. Next, we consider possible types of uncertainties in decision aiding.

2.2.1 Decision Aiding under Uncertainty

When modeling a decision situation that is a fragment of reality, it is not possible to avoid uncertainty, which to some extent is included in each model. This issue should not be omitted and both the analyst and the DM should be aware of the presence of uncertainty and search for ways to handle it. Analyzing and structuring the problem may help to specify and better understand the nature of uncertainty, and in such a way the uncertainty may be reduced. There are different types of uncertainty and different ways to classify it. Belton and Stewart distinguish internal and external types of uncertainty [16].

Internal uncertainty is embedded into the structure of the DM's preferences as well as into the MCDA method considered. A part of this type of uncertainty is resolvable. For instance, there can be difficulties with understanding the meaning used for the description of criteria and interpretation of values on the scales of each criterion. The details in terms of precision and selection of appropriate scales of values is subject of discussion but still may be unresolved in some cases. The alternatives that should be or should not be included into the set of given alternatives should be discussed for compact but coherent formulation of the problem. When estimating alternatives with qualitative scales of criteria values another problem may appear. Verbal information, in some sense, is uncertain. In formal methods, the most widely-spread way of working with verbal information is its transformation to some numerical analogue. However, any transformation evokes a loss of some information. Consequently, working with transformed information leads to accumulating inaccuracy. This important feature should be taken into account.

As it was already quoted, "a problem well-stated is a problem half-solved" (John Dewey, philosopher). That is why the step of structuring the problem is very important. The structure is developed at the initialization step of decision aiding process based on the available initial data and requirements to the resulting data. If there are several persons participating in the process of decision aiding (e.g., the analyst and the DM), it is important to fix the vocabulary of terms and definitions common for them. At the next step, where method is selected, another type of uncertainty may appear, such as uncertainty in the parameters of the specific model. Understanding parameters of the model selected for the decision aiding is very important, since they may have a different meaning in different methods (for instance, see Section 5.1.9). This type of uncertainty may be resolved by a more precise analysis of requirements and given data. Another kind of uncertainty is connected with the human nature. Indeed, the DM may not be sure about his or her own preferences, which also may evolve over time.

External uncertainty about decision areas or environment depends on the knowledge of the DM about consequences of his or her decisions. Uncertainty about decision areas is related to the interconnection between different decisions made in the problem area. For example, in a classification problem, it may be about the influence of the assignment of one alternative to the class if the assignment of another alternative to another class is done. The knowledge of the DM about the behavior of the environment is usually uncertain and may be limited.

In this work, we consider aspects of uncertainty in the framework of classification problems that may be resolved by the DM by means of better structuring the problem, choosing the method that best suits the considered situation as well as helping the DM to understand his or her own preferences.

In decision aiding, there are several basic approaches for the development of decision aiding. Next, we consider assumptions that they impose on the behavior of DM.

2.2.2 Decision Aiding Approaches

The inference of the DM's preferences, the so-called preference elicitation, can be done in different ways. These procedures depend on the "model of human rationality", called a rationality model, a tool that is able to translate informal information (that is naturally ambiguous) to a more formal form that still contains ambiguity, but it is represented in an unambiguous way (for details, see [156]).

The first models of human rationality developed in the framework of classical decision theory [159] assume that the DM behaves as a rational person according to the following principles or norms of rational or economical behavior. It is assumed that the DM always maximizes his or her expected performance, he or she can always define all parameters needed for the model, and has enough knowledge for making a decision. Such models include uncertainty in the form of probabilities. However, such assumptions are far from what happens in the real life.

Recently, a lot of research has been done in studying the nature of human's rationality (see survey in [156]). Indeed, Simon in his work [143] studied the behavior of people when making decisions and introduced the bounded rationality principles. He proves that, in practice, there are always unpredictable circumstances and incompleteness of information that is available (just because it is impossible to include in the decision aiding model all the preferences of the DM, not to think about multiple DMs). The DM usually behaves according to the local situation and searches not for the best decision ever but for a compromising and acceptable one at the moment of decision making. Finally, the resources and time of a DM are always limited. These findings gave birth to a new understanding of the model: the model developed in the framework of a decision aiding process and within a person.

A decision aiding approach characterizes how the decision aiding process is conducted. Thus, several approaches to decision aiding originating from different ways to construct the decision aiding model (see Section 2.1.1) depending on the assumed model of human rationality can be identified. These include normative, descriptive, prescriptive and constructive approaches [33], [156].

A *normative approach* assumes that the DM is rational and his or her behavior is described by norms of rational behavior established a priori. The rationality model consists of adaptation of DM's ways of expressing preferences to the specified formal model. This approach assumes that the rationality model is objective and exists separately from the particular DM, in addition, it cannot be changed (also during the decision aiding process). If the DM does not follow the principles of rational behavior, he or she makes mistake; also uncertainties are considered as probabilities in the model (that may not always be the case in real-life situations). Even though this approach contradicts with principles of Simon's bounded rationality discussed earlier, it is still widely-used and may be appropriate in some situations.

A *descriptive approach* observes the behavior of a real DM and identifies his or her cognitive profile. Then, the formal model that better suits such a behavioral profile is selected. This approach assumes that other DMs act similarly when facing the same problems. When compared to the normative one, the descriptive approach is more subjective and satisfies principles of bounded rationality.

A *prescriptive approach* discovers a rationality model for a given DM from his or her answers to a set of questions related to the preferences about the decision situation. Even though the approach considers the whole decision aiding process (see Section 2.2), it concentrates on fixing the model to be used by the DM in a particular situation. This approach supposes that each DM has his or her own rationality model that cannot be defined separately from him or her. Thus, it is even more subjective than the descriptive approach. The DM may be inconsistent when expressing his or her preferences. However, all inconsistencies should be detected and then resolved by the DM.

A *constructive approach* builds a rationality model for a given DM from his or her answers to a set of questions related to the preferences about the decision situation. This approach concentrates not only on discovering and building the model but also on structuring and learning of the DM's own preferences. The DM may be inconsistent with his or her preferences, and this is not considered to be a problem. On the contrary, inconsistencies are seen as a source for discussion and improvement of the model that may evolve during the decision aiding process. Finally, there is no requirements for removing inconsistencies and the final model tries to find a solution assuming presence of incomparabilities and intransitive judgements.

As can be seen, the first approaches where developed for more general situations and aimed to be as objective as possible. On the other hand, the most recent approaches pursue studying a particular situation and behavior of a particular DM and allow him or her to learn. Thus, the latest trends in the development of decision aiding approach is to move towards subjective models developed in the framework of a decision aiding process and within a person.

Finally, one should mention that this variety of decision aiding approaches is developed for different situations, and there is no special preference to any of them. When selecting the decision aiding approach to be used the analyst has to think about two questions [33]: "Where the rationality model comes from?" and "How is the model obtained?" The answer to the first question allows selecting more general and objective approaches (normative and descriptive) or subjective and specific approaches (prescriptive and constructive). The answer to the second question distinguishes the models that are postulated (normative approach), or derived from observations (descriptive approach), from models that trying to discover the models (prescriptive approach) or to build the models (constructive approach) based on a dialog with the DM.

Even though MCDA methods developed in the framework of decision aiding theory (see Section 2.2.3) may assume one of the decision aiding approaches when developing a rationality model, the methods are independent from the approaches used and vise-versa [33]. For instance, it is possible to use an outranking based preference aggregation procedure (that is explicitly claimed have been developed within constructive philosophical justifications [140]) within a normative approach. Next, we consider some widely-used MCDA theories that have been developed.

2.2.3 MCDA Theories

A survey of the MCDA literature shows that the utility theory and the outranking approach are the two most widely-used directions in MCDA theory.

A utility theory (UT) assumes that a DM is at least unconsciously maximizing some utility function during the decision aiding process. This function represents a global aggregation of utilities of alternatives on each criterion. Thus, it is necessary to define the form of criteria functions. The preferences of the DM are expressed in the form of criteria weights. The weights in UT represent trade-offs between criteria. In order to represent uncertainties, UT uses probabilities (which other MCDA approaches consider as a shortage). UT can work with different aggregation forms for utility functions, the main aggregation forms being additive, multiplicative and distributional. The approach considers all the alternatives defined within a problem as comparable and independent. Since von Neumann and Morgenstern [159] who developed UT, many methods have appeared in the framework of this approach, and include Multiattribute Utility Theory (MAUT) [80] and Analytical Hierarchical Process (AHP) [141].

An outranking approach uses DM's preferences in order to compare pairs of alternatives. Based on the pairwise comparison, a complete or non-complete preorder is built. The methods based on this approach use the concept of outranking relation (see Section 2.1.5). The main idea is to estimate the outranking degree of one alternative over another one, that is, to show that there are enough arguments to consider "one alternative to be at least as good as another alternative" and there are not enough arguments in the opposition to such an assertion. In comparison to UT, outranking methods can take into account incomparability of alternatives as well as have a noncompensatory character. Two alternatives are considered incomparable if they are not indifferent to each other and it is not possible to say which of the two alternatives is better or worse. Compensatory models, such as UT, consider the trade-offs between criteria that are usually expressed by criteria weights. In the outranking approach, on the other hand, there is no technique for estimating trade-offs between criteria. Weight is used to represent relative importance of a criterion when compared to other criteria in terms of votes that support or oppose the assertion "one alternative is at least as good as another alternative".

The preference information for different outranking methods may be expressed in weights and through preference, indifference and veto thresholds. Currently, there are many MCDA methods developed according to the principles of this approach, and include ELECTRE [139], [140] and PROMETHEE [158] families of methods.

The two above-mentioned theories require some preferential information in the form of parameters of the model to be defined by the DM. However, it may be difficult for the DM to understand the meaning of the parameters of a particular model and/or express his or her preferences. That is why, recently, two other MCDA directions have been intensively developed: preference elicitation procedures and rules inferring also called here decision rule approach [172].

Preference elicitation procedures are indirect techniques that do not require the DM to express his or her preferences in the form of parameters of a specific model, even though they assume the form of the model to be known (e.g., utility theory or outranking approach). One way of the preference elicitation is used with the assumption that there are decisions available made during the past experience by the DM or that he or she may make some decisions for realistic alternatives not considered in the set of given alternatives, or decisions made for a subset of alternatives from the set of given alternatives. Based on such available decisions, the parameters of the specific model are elicited. A so-called preference disaggregation approach has been developed that elicits preferences in the form specified by the model (either the utility theory or the outranking approach) with one of the mathematical programming techniques. A number of methods have been developed in the framework of preference disaggregation approach assuming utility theory models, such as UTA [71], UTADIS [37], [72] and based on an outranking approach, such as ELECTRE TRI Assistant [119]. On the other hand, SMAA [87], [153] performs simulations of the parameters of a specified model based on the Monte-Carlo simulation. Usually, with these methods, there is no need for specifying any information (such as decisions earlier made by the DM), but, if some partial information about the parameters of the model (e.g., in the form of intervals of values or their distribution) is available, it can be used in the SMAA methods.

A decision rule approach is a technique that describes the behavior of the DM by means of "*if* ..., *then* ..." rules or decision trees [172]. This idea is adapted from artificial intelligence, where a similar approach is used, for instance, in expert systems. It also assumes availability of the decisions made by the DM in the past or readiness of the DM to make some decisions for realistic alternatives not considered in the set of given alternatives or for a subset of given alternatives. Several methodologies have been developed in the framework of the decision rule approach, such as verbal decision analysis [99], [100], fuzzy sets [15], [166], and rough sets [56]-[60].

Now that the most important concepts and terminology of MCDA have been introduced, we have a sufficient background for considering classification within the MCDA framework.

2.3 MCDA and Classification

The rest of this work concentrates on classification. After describing several classification problems, we investigate different types of methods for solving classification problems with the help of MCDA approaches.

In many situations, the DM faces the problem of grouping alternatives into homogeneous classes: this we call a classification problem. Usually, a classification task appears in diagnostics, when the DM assigns each alternative from the set of given alternatives to one or several classes. For example, a doctor arrives at a diagnosis for a patient, an engineer establishes the type of problem when a machine is broken down, and a biologist classifies a new plant to one of the existing classes of plants or defines a new classification for it. The methods that solve such problems should be transparent to allow the DM to perform effective decision making.

2.3.1 Types of Classification

The central concept of a classification problem is class. A *class* is a collection of alternatives with similar properties or even values for the same properties, when compared to alternatives in other classes. The classes in a problem can be predefined and well-described; it is also possible that information about the classes is absent. If the classes are known beforehand, then the problem is called a classification problem. On the other hand, when there is no description of the classes or even the number of classes is absent, the problem is known as a clustering problem.

A *clustering problem* (see Figure 1) is also called an unsupervised learning problem. In this case, classes are not defined and sometimes even the number of them is unknown a priori. On solving a clustering problem, it is necessary to locate in one cluster alternatives that are more similar to each other, when compared to the alternatives in the other clusters. Clustering methods are differentiated by the type of criteria used for optimal clustering, the distance or the similarity measure, and the searching algorithms used [64], [73], [109].

The *classification problem* is also called a supervised learning problem. The classification problem can be considered as a particular case of a clustering problem when additional information about the number of classes and their structure is known a priori. There are two types of classification problems: *nominal*, when classes are not ordered, and *ordinal*, when they are ordered according to the preferences of the DM. The ordinal classification can be considered as a particular case of the nominal one, when additional information about the order of the classes is available. Utilizing such information about the order of classes is specific for the MCDA methods, and it can improve and/or simplify the modeling process in some cases.

There are different definitions of classification problem types in the MCDA literature. For instance, Perny [131] describes classification as follows: "Given

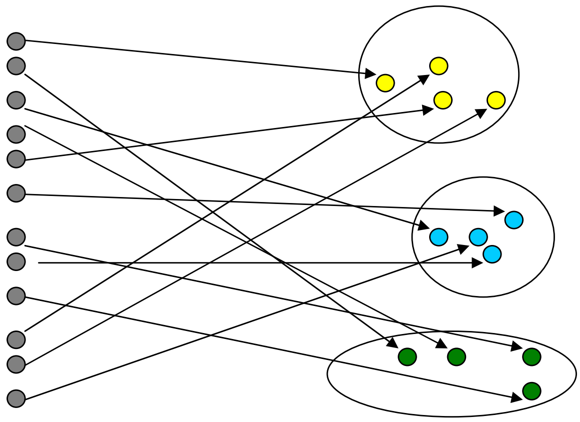

FIGURE 1 Clustering

a set of alternatives evaluated with respect to several criteria and a list of predefined categories characterized by specific reference points in the space of criteria, the multicriteria assignment problem is to evaluate the membership of each alternative in each category. If a value judgment is associated with each category the assignment problem amounts to evaluate the intrinsic quality of each alternative". In this work, categories are classes, reference points are reference or boundary alternatives for nominal and ordinal classification types, respectively, and a multicriteria assignment problem is called as ordinal or nominal classification for the case when classes are ordered and not, respectively (see [37] and [140] for alternative definitions of classification problem).

In a nominal classification, it is necessary to assign alternatives to classes predefined in some way. Usually, the number of classes is known a priori. In the nominal classification problem, the classes may be (but not necessary are) predefined with reference alternatives. A reference alternative is an alternative typical for the class. In Figure 2, the not yet classified alternatives are shown as grey circles (on the left); after classification they appear in three different classes as yellow, blue and green circles, respectively; and with the reference alternative as a red circle for each of the class.

In an ordinal classification, it is necessary to assign alternatives to ordered classes predefined in some way. Again, the number of classes should be known a priori. An ordinal classification problem may use (but not necessary does) boundary alternatives for constraining classes. In Figure 3, instead of a reference alternatives for each class, we have dark blue circles that represent boundary alternatives between each pair of classes. The boundary alternative is an alternative that lies in the border between two classes and, changing values on at least one criterion, will move this alternative to a neighboring class. The boundary alternatives are ordered according to the classes, which they constrain.

A classification procedure consists of two steps. In the first step, a model of the problem is built. During this step, the structure of the classes is defined in some way, for instance, with the reference or the boundary alternatives for each

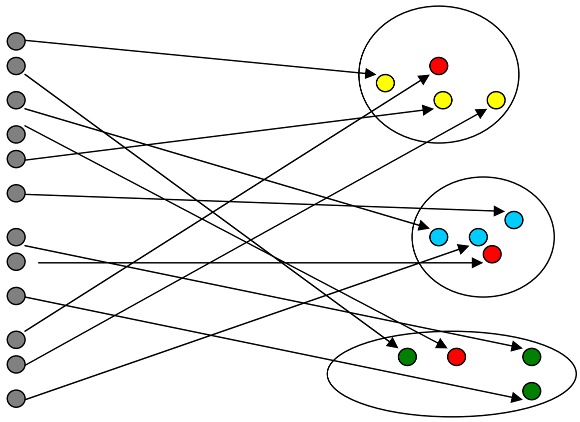

FIGURE 2 Nominal classification

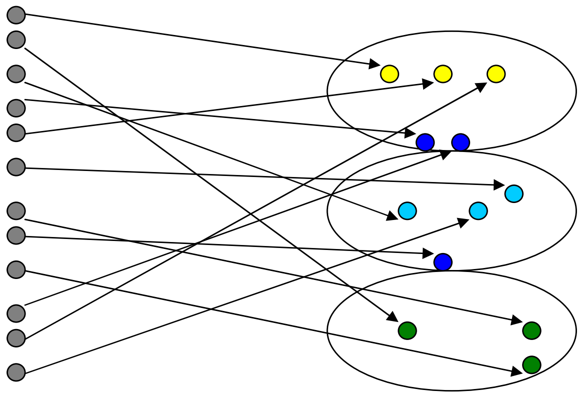

FIGURE 3 Ordinal classification

class. In the second step, the model is tested, and then it is utilized for assigning new alternatives to one of the predefined classes. As already mentioned, the classification problem with MCDA is also called a *multicriteria assignment problem* [10] and ordinal classification is often called a *sorting problem* [37] (see also [26], [172]).

In what follows, we concentrate on classification problems. In the next section, a mathematical formulation of a classification problem is considered.

2.3.2 Mathematical Model of MCDA Classification

The mathematical formulation of a classification problem can be described as follows: it is necessary to assign the set of alternatives $X = \{x_1, \dots, x_m\}$ evaluated on the set of criteria $G = \{g_1, \dots, g_n\}$ to one class from the predefined set of classes (ordered or not) $L = \{l_1, \dots, l_s\}$. If an assignment to several classes is allowed, it is specified. An estimation of the alternative x_i on the criterion g_j is denoted as $g_j(x_i)$. Additional notation specific to each method will be explained while presenting these methods. Next, we define some specific cases of nominal and ordinal classification when additional information about reference or boundary alternatives, respectively, is available.

If reference alternatives are available, they may be used for predefining classes in the nominal or ordinal classification. A *reference alternative* is an alternative that most completely characterizes the class. The set of reference alternatives is defined as $B = \{b_1, \ldots, b_s\}$, where s is the number of classes. There can be several reference alternatives for each class. In this case, the set of reference alternatives is defined as $B = \{b_1^1, \ldots, b_1^{t_1}, \ldots, b_s^1, \ldots, b_s^{t_s}\}$, where t_1 and t_s are the numbers of reference alternatives for the classes l_1 and l_s, respectively.

After the definition of classes, the actual classification procedure can be considered: each alternative from the set X should be assigned to one of the classes from the set L based on the rule of assignment, or the chances for each alternative to be assigned into each class are estimated. In case the nominal classes are predefined with reference alternatives, the assignment rule is the following: an alternative is assigned to a class if it is equivalent or roughly equivalent to at least one of the reference alternatives of this class. Later, we discuss what "equivalent" or "roughly equivalent" mean in methods.

If boundary alternatives are available, they may be used for ordinal classification. A *boundary alternative* belongs to a class, but changing the value on at least one criterion will move this alternative to a neighboring class. A set of boundary alternatives is defined as $B = \{b_0, \ldots, b_s\}$, where s is the number of classes, and b_{q-1} ($q = 0, \ldots, s$) is the lower bound and b_q is the upper bound of the class l_q, thus boundary alternatives separate classes. In case of several boundary alternatives, for each class the set of boundary alternatives is defined as $B = \{b_0^1, \ldots, b_0^{t_1}, \ldots, b_s^1, \ldots, b_s^{t_s}\}$, where b_q^h ($h = 1, \ldots, t_q$, $q = 0, \ldots, s$) is one of several boundary alternatives between classes l_{q-1} and l_q. In this case there could be several upper boundary alternatives and several lower boundary alternatives for each class. In case the ordered classes are predefined with boundary alternatives, the classification is realized according to the following rule: an alternative is assigned to a class if it is located between the lower and the upper boundary alternatives of a class.

In the next chapter, we consider different MCDA methods developed recently for ordinal and nominal classification types. The survey provided tries to cover the most widely-spread methods for classification within the MCDA framework.

3 MCDA METHODS FOR CLASSIFICATION

Originally, MCDA methods were developed for choice and ranking types of problems, and, thus, most of them are based on the measurement of the degree of preferences between alternatives from the given set. That was the reason for the ordinal type of classification problem to appear in MCDA as a natural relaxation of a ranking problem for the condition that at each rank level several alternatives can be assigned. Later, however, the MCDA methodology has been extended for nominal type of classification as well as for clustering problems. In this chapter, we consider the methods for ordinal and nominal types of classification problems.

At the beginning of the decision aiding process, the analyst together with the DM structures the problem at the initialization step of the decision aiding process (see Section 2.2). For the classification problem, we have to define the set of alternatives to be classified, the set of criteria and/or attributes, on which the alternatives are evaluated, as well as the set of classes. It is important to define the scales of criteria and/or attributes and how the criteria and/or attributes values are measured at each scale. The criteria and/or attributes can have numerical or verbal values. In turn, the numerical values may have a form of exact values, intervals, or distributions of criteria values. The requirements for the set of criteria are checked (see Section 2.1.4). It should be also specified if there is order between the classes or not, and if it is possible to define the classes with boundary or reference alternatives (for ordinal and nominal classification types, respectively).

If there is no possibility to define the reference or boundary alternatives, we have to ask the DM if he or she is able to specify some assignment examples (that may be decisions the DM used to make in the past or decisions made for realistic alternatives not considered in the set of given alternatives). Otherwise, we have to find a way to ask the DM to classify some (but not all) alternatives from the set of given alternatives. For ordinal classification problems, if the boundary alternatives are available, their order should correspond to the order of classes that they constrain. Indeed, if we have the following order of classes according to the preferences of the DM:

$$l_s \mathrm{~P~} l_{s-1} \mathrm{~P~} \cdots \mathrm{~P~} l_1,$$

then the boundary alternatives are ordered as follows:

$$b_s \mathrm{~P~} b_{s-1} \mathrm{~P~} \cdots \mathrm{~P~} b_1.$$

The case of several lower and several upper boundary alternatives for each class is defined by analogy with only a difference that in the set of lower (or upper) boundary alternatives for the same class the relation of incomparability is established:

$$b_s^1 \mathrm{~J~} \cdots \mathrm{~J~} b_s^{t_s} \mathrm{~P~} b_{s-1}^1 \mathrm{~J~} \cdots \mathrm{~J~} b_{s-1}^{t_{s-1}} \mathrm{~P~} \cdots \mathrm{~P~} b_1^1 \mathrm{~J~} \cdots \mathrm{~J~} b_1^{t_1}.$$

We have also to consider in what form the final classification should be presented, that is, weather each alternative should be assigned into exactly one class or weather it would be enough to provide a probability of the alternative to be assigned to each class. Then, the MCDA method is selected with respect to the information and structure of the problem defined at the initialization step.

There is a large variety of methods for classification in MCDA. The diversity of the methods is due to the different types of classification problems, the initial information available and requirements to the resulting one. The opinion of the DM regarding the decision aiding process should also be taken into account. The analyst should select the method that is suitable for the DM and that corresponds to his or her knowledge and system of preferences.

It can be said that the selection of the MCDA method is an MCDA problem itself (by analogy with the selection of multicriteria optimization method in [107]). Some recommendations for the selection of an MCDA method for classification are provided in Section 5.2.

The most widely-used MCDA methods for classification have been mainly developed within the MCDA theories presented in Section 2.2.3. Literature reviews on the MCDA methods for classification can be found in [26], [37], [161], [172].

In this work, we categorize the MCDA methods for classification discussed in more detail in a similar way as it is done in [172], more specifically, according to the assumption about the possibility to describe the behavior of the DM by means of some formal model involving some parameters and according to the way the preferences of the DM are elicited (see Figure 4). Two types of MCDA methods for classification are considered: the first type can be called *parameter-based* methods, and the second type *parameter-free* methods. For the first type, it is possible to construct a formal model of the DM's behavior based on some parameters. Here, two types of models are defined: *utility function-based* and *outranking relation-based*. Parameter-free methods contain so-called decision rule methods [172], which assume that the behavior of the DM is too complex to be described by a formal model based on some parameters, but there are possibilities to define it with a judgemental model based on a set of rules. The parameter-based methods are also categorized further according to the way the parameters of the model

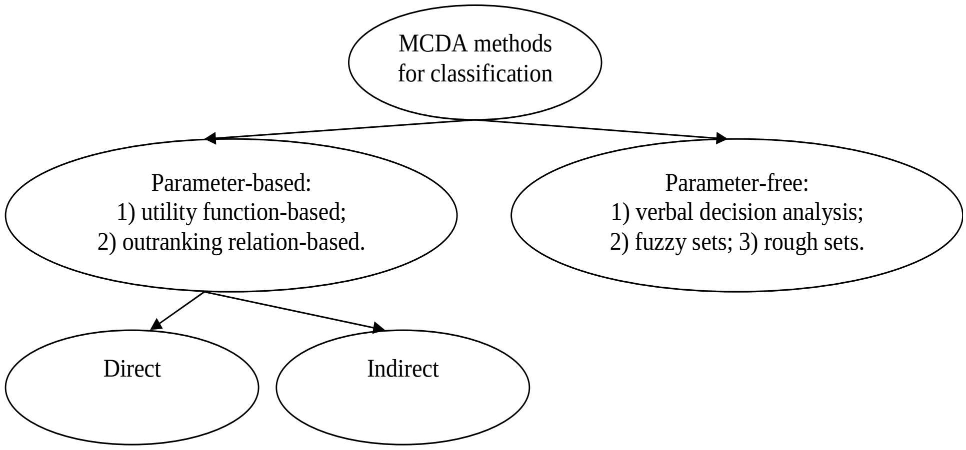

FIGURE 4 Categorization of MCDA methods for classification

are elicited on *direct* and *indirect* methods. With the direct methods, the DM provides all parameters of the model, while with the indirect ones, the parameters are elicited based on the so-called assignment examples that are decisions made by the DM in the past or made on a set of realistic alternatives not considered in the set of given alternatives or a subset of the set of given alternatives.

Thus, three types of MCDA methods for classification are considered. The first type is direct parameter-based methods, and the second type is indirect parameter-based methods. There can be utility function-based and outranking relation-based direct and indirect methods. The third type is parameter-free decision rule methods. Let us consider these types in more detail.

Even though outranking methods and utility theory approaches assume principally different formal models, in both cases the parameters of the model should be well-defined. Usually, this means that the DM should be able to express his or her preferences in the form of parameters of some formal model. That is why these methods are called direct parameter-based methods.

Most of the MCDA methods developed for ordinal classification are direct outranking based methods. Among them is the first MCDA method developed for ordinal classification, called Trichotomic Segmentation [111], [136]. This method belongs to the ELECTRE family of methods and is based on the concordance and non-discordance principle [140]. It is limited to three classes only. Later, it was extended to the N-TOMIC method [105] and ELECTRE TRI method [116], [165] with an arbitrary number of ordered classes. The utility based method AHP [141] has also been adapted for classification problems [148].

A number of direct parameter-based methods have been developed within the outranking approach for the nominal type of classification. These include the MC Filtering method [131] and PROAFTN [10], [12].

In this work, in order to provide detailed presentation of the outranking methodology we select the most widely-used direct outranking relation-based method for ordinal classification, that is, ELECTRE TRI (see Section 3.1.1). Two other direct outranking relation-based methods, MC Filtering (for ordinal and nominal classification types) and PROAFTN (for nominal type of classification), are also considered in more detail. One more method for nominal classification discussed in this work is TRINOMFC [101] that is also an outranking based method that, however, stems from another family of methods, PROMETHEE [23].

The so-called indirect parameter-based methods, which also assume a particular formal model to be used, do not suppose that the DM is able or have to define all the parameters of the model. These methods try to reduce the cognitive load of the DM and to alleviate the inconvenience experienced by the DM when specifying the parameters of a model in a particular form. In some of these methods, the parameters of the model are elicited from a set of assignment examples made by the DM in the past or made on the set of realistic alternatives not considered in the set of given alternatives or on a subset of given alternatives. The idea has been adapted from the machine learning and training or learning from assignment examples. Based on such information, the model is constructed and applied for the classification of the given set of alternatives.

A number of indirect utility function-based methods have been developed in the framework of the preference disaggregation approach [37]. The preference disaggregation approach elicits the DM's preferences from the assignment examples based on one of the mathematical programming techniques. The following methods have been developed in the framework of preference disaggregation approach: UTADIS [37], PAIRCLAS [38] and MHDIS [35]. In this work, we consider the UTADIS method in more detail (see Section 3.3.1). For the ELECTRE TRI method, preference elicitation procedures have been developed, among them DIVAPIME [113], SRF [46] and ELECTRE TRI Assistant [119], [124]. In this work, we consider two of them: DIVAPIME and ELECTRE TRI Assistant (see Section 3.2). ELECTRE TRI Assistant also uses preference disaggregation approach.

Another way of indirect preference elicitation has been suggested in SMAA methods [87], [153]. In these indirect parameter-based methods, the preferences of the DM are simulated by means of Monte-Carlo simulations. We consider in more detail the SMAA-TRI method [151] that uses an outranking based ELECTRE TRI model (see Section 3.4). The new SMAA-Classification method [164] that uses SMAA for simulation of parameter values based on the distance function is presented in Section 4.1.

We define the third type of MCDA methods for classification as parameter-free decision rule methods. These methods assume the structure of the DM's preferences to be too complex for definition in terms of some formal model. Instead, the behavior of the DM is described in a symbolic or judgemental form by means of decision rules. This approach is adapted from machine learning techniques, such as expert systems. The idea is to search for the classification rules that, like in preference disaggregation approach, are based on a set of assignment examples. A number of methods has been developed in the framework of this approach, such as verbal decision analysis [99], [100], fuzzy sets [15], [166], and rough sets [56]-[60]. In this work, we consider some methods from all the three approaches (see Sections, 3.5, 3.6, 3.7). We have also developed a new MCDA classification method, called Dichotomic Classification, (see Section 4.2) in the framework of verbal decision analysis.

While both fuzzy sets and rough sets assume a set of assignment examples to be available, in verbal decision analysis such information is absent a priori. The verbal decision analysis methods are interactive. During the dialog, the DM has to classify some alternatives selected by the method, but not all. In the framework of verbal decision analysis, the following methods have been developed: ORCLASS [100], SAC [95] and a new Dichotomic Classification method [163] (to be presented in Section 4.2).

In this work, we discuss in more detail methods published in English before the summer 2007 in widely-accessible journal articles. Some other methods for ordinal and nominal classification types are not covered by the present survey. These include STEPCLASS [50], methods based on fuzzy integrals [110] and TOMASO [103], FlowSort [122], and screening methods [26]. There are also classification approaches discussed in works [1], [5], [20], [21], [82], [142] and applications presented in [3], [7], [25], [125]. We have not described these works in more detail because they are based on different philosophies than the methods presented here. However, we include several of them in Figures 17 and 16 in Section 5.2, for completeness, in order to show where they would be located in the decision trees when selecting an MCDA method for classification.

Each approach has advantages and disadvantages; there is no single one that suits every possible situation. That is why the available initial information and requirements for results as well as the nature of the classification problem and the DM's opinions about the suitability of the method to his or her own way of thinking should be considered when selecting a method. After comparing features of the presented methods in Section 5.1, we wish to give some recommendations to the analyst and DM for selecting an MCDA-based method for classification (see Section 5.2).

As has been mentioned earlier, the first classification methods in MCDA stem from the ranking problem and have been developed for ordinal classification problems. In ordinal type of classification methods, additional information about the order between classes is assumed to be available. Usually, these methods estimate preferences between alternatives. On the other hand, later, the nominal type of classification methods has become studied. These methods not utilize any information about the order on the set of classes, and for them estimation of the indifference or similarity between alternatives is more natural. This is the reason for us to describe ordinal methods first and then to present nominal ones in next sections. Methods for both types of classification problems may stem from the same MCDA theories (see Section 2.2.3) and may have some similar features; in this way they are connected. In what follows, some of these connections are discussed when describing methods to point out similarities.

3.1 Outranking methodology

In the outranking approach, the idea was to develop a theory that would allow comparing a pair of alternatives while taking into account some level thresholds that determine when to be indifferent or when to prefer one over the other on the same criterion. At the same time, the conflicting nature of the multicriteria problems should also be paid attention to, and two alternatives should be allowed to be incomparable on the same criterion. In such a way, the methods have a non-compensatory nature that means an absence of trade-off between criteria.

For a historical overview on the development of outranking approaches, we refer to [23], [45], [104], [135], [140]. Here, we only briefly mention that the ELECTRE family is most developed group of outranking methods. The first outranking-based ELECTRE method was developed in 1965 by Roy [135] in order to choose the best alternative from the set of given alternatives. Then, ELECTRE Iv appeared using additional information about veto threshold. The next version, ELECTRE IS, allowed initial information to be imprecise and considered pseudo-criteria. This version is still in use for the choice type of MCDA problems. The subsequent versions, ELECTRE II and ELECTRE III, were developed for ranking. The latter one uses pseudo-criteria and fuzzy binary outranking relations and is the current version for the ranking type of MCDA problems. One more version, ELECTRE IV, exists for problems where the ranking has to be done without taking into account the relative importance of criteria.

As mentioned earlier, the first outranking-based classification method, Trichotomic Segmentation, [111], [136] was developed for sorting problems with three classes. Later, this method was extended to an arbitrary number of classes in the N-TOMIC [105] and ELECTRE TRI methods [119], [165]. Before considering ELECTRE TRI in more detail in the next section, it is worth to say that the development of this method continues and we consider the latest trends in Section 3.1.2.

There have been attempts [47] to develop the classification method in the framework of another outranking approach, the PROMETHEE-based method, introduced by Brans [23] and also described in [158]. The latest one is the Flow-Sort method [122] that allows nominal and ordinal classification. There are still possibilities for the development of the classification methods in the framework of other outranking approaches, for instance, within an outranking method based on the stochastic dominance relation [104].

3.1.1 ELECTRE TRI

ELECTRE TRI is a method for ordinal classification. The basis for the ELECTRE TRI method [45], [119], [165], as well as for all the methods from the ELECTRE family, is estimation of the outranking relations between pairs of alternatives. As defined in Section 2.3.2, in the classification problem, there is a set of given alternatives to be classified X estimated on a set of criteria G to be assigned into a set of ordered classes L predefined by the set of boundary alternatives $B = \{b_0, \ldots, b_s\}$. Each class is constrained by two (upper and lower) boundary alternatives. Boundary alternatives separate classes in such a way that the upper bound b_q of the class l_{q-1} is the lower bound of the class l_q $(q = 1, \ldots, s)$. Changing values on at least one criterion moves the boundary alternative to a neighboring class. For solving a classification problem, the method estimates the outranking relation for each alternative $x_i \in X$ $(i = 1, \ldots, m)$ to be classified and each boundary alternative b_q between classes l_{q-1} and l_q by calculating the outranking index. We assign the alternative x_i to the class l_q if it is preferred to the lower boundary alternative b_{q-1} of the class, and if the upper boundary alternative b_q of the class is preferred to this alternative.

For the calculation of the outranking index, it is necessary to elicit the following information from the DM: a set of alternatives to be classified and a set of criteria, on which the alternatives are evaluated with a scale of quantitative values for each criterion. We ask the DM to specify the number of classes as well as their order, according to his or her preferences; for example, the first class is more important than the second one and so on. He or she should also define the upper b_q and lower b_{q-1} boundary alternatives for each class l_q. In addition, the ELECTRE TRI method requires the following information for each criterion g_j $(j = 1, \ldots, n)$: indifference $q_j(\cdot)$, preference $p_j(\cdot)$ and veto $v_j(\cdot)$ thresholds as well as weight w_j; and cutting level λ. The indifference threshold $q_j(\cdot)$ indicates the largest difference between two alternatives on the criterion g_j such that they remain indifferent for the DM. The preference threshold $p_j(\cdot)$ indicates the smallest difference between two alternatives such that one alternative is preferred to the other one on the criterion g_j. The veto threshold $v_j(\cdot)$ defines the smallest difference between two alternatives on the same criterion g_j that shows incomparability of these two alternatives. The thresholds should satisfy following constraint $v_j(\cdot) > p_j(\cdot) > q_j(\cdot)$. The weight w_j of criterion g_j indicates the relative importance of criterion when compared to other criteria in terms of votes that support or oppose the assertion "one alternative is at least as good as another alternative". The cutting level λ shows the smallest value of the outranking index that is sufficient for considering an outranking situation between two alternatives.

The outranking relation is verified by two conditions: concordance and non-discordance with regards to thresholds and weights defined by the DM. The first condition requires preference of the alternative x_i over the boundary alternative b_q on the majority of criteria and the second one demands the absence of strong opposition to the first condition in the minority of criteria. We compute two partial indices for each criterion: concordance $C_j(x_i, b_q)$ and $C_j(b_q, x_i)$, and discordance $D_j(x_i, b_q)$ and $D_j(b_q, x_i)$. They allow calculating the outranking indices $S(x_i, b_q)$ and $S(b_q, x_i)$. The assignment procedure consist of a comparison of outranking indices to the cutting level λ. There are two assignment procedures: pessimistic and optimistic. The pessimistic procedure starts with the comparison of an alternative to the lower bound of the highest class: we calculate the outranking indices and compare them to the cutting level. The optimistic procedure works in the same way but begins with the comparison of an alternative to the

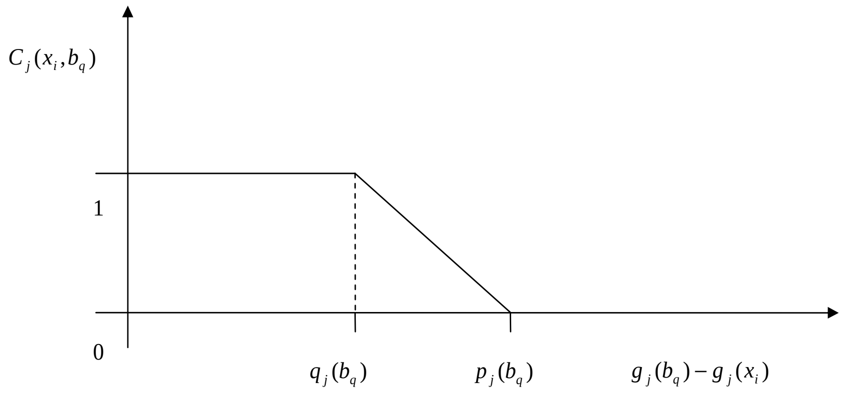

FIGURE 5 ELECTRE partial concordance index $C_j(x_i, b_q)$

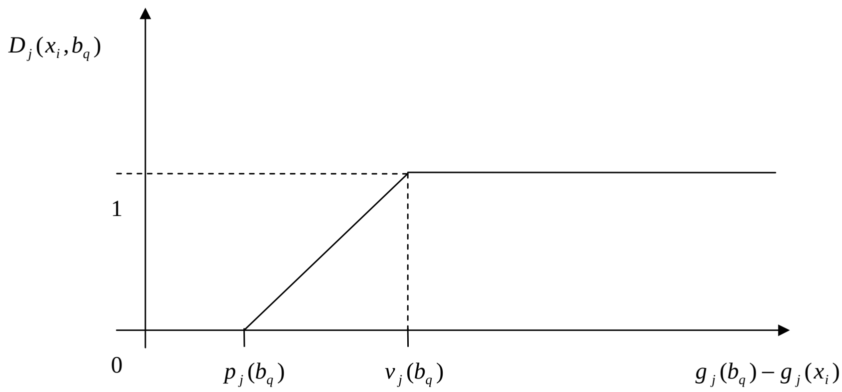

FIGURE 6 ELECTRE partial discordance index $D_j(x_i, b_q)$

upper bound of the lowest class. In order to assign the alternative x_i to the class l_q according to the pessimistic procedure (or to the class l_{p+1} with regards to the optimistic one), the alternative should have the value of the outranking index $S(x_i, b_q)$ bigger than the cutting level λ and value of the index $S(x_i, b_{q+1})$ smaller than λ. The DM can select one of the two assignment procedures or apply both of them. Next, we can discuss the algorithm of the ELECTRE TRI method.

The algorithm of the ELECTRE TRI method

The procedure of classification with ELECTRE TRI consists of two parts: construction of outranking relation (I) and utilization of this relation for the assignment of alternatives to classes (II).

Part I. Let us construct the outranking relation $x_i S b_q$ for each alternative x_i ($i = 1 \ldots, m$) to be classified and each boundary alternative b_q ($q = 0, \ldots, s$) with the following steps:

Step 1 Calculate the partial concordance indices $C_j(x_i, b_q)$ and $C_j(b_q, x_i)$ for each criterion g_j ($j = 1, \ldots, n$). Because a maximization problem is under consideration, the criterion g_j has an increasing direction of preference. The partial concordance index $C_j(x_i, b_q)$ is computed as follows (see Figure 5):

$$C_j(x_i, b_q) = \begin{cases} 0, \text{ if } g_j(b_q) - g_j(x_i) \geq p_j(b_q), \\ 1, \text{ if } g_j(b_q) - g_j(x_i) < q_j(b_q), \\ \frac{p_j(b_q) - g_j(b_q) + g_j(x_i)}{p_j(b_q) - q_j(b_q)}, \\ \text{ if } g_j(b_q) - p_j(b_q) < g_j(x_i) \leq g_j(b_q) - q_j(b_q). \end{cases}$$

The partial concordance index $C_j(b_q, x_i)$ as follows:

$$C_j(b_q, x_i) = \begin{cases} 0, \text{ if } g_j(x_i) - g_j(b_q) \geq p_j(x_i), \\ 1, \text{ if } g_j(x_i) - g_j(b_q) < q_j(x_i), \\ \frac{p_j(x_i) - g_j(x_i) + g_j(b_q)}{p_j(x_i) - q_j(x_i)}, \\ \text{ if } g_j(x_i) - p_j(x_i) < g_j(b_q) \leq g_j(x_i) - q_j(x_i). \end{cases}$$

Step 2 Define the overall concordance indices $C(x_i, b_q)$ as an aggregation of partial concordance indices:

$$C(x_i, b_q) = \frac{\sum_{j=1}^{n} w_j C_j(x_i, b_q)}{\sum_{j=1}^{n} w_j},$$

$$C(b_q, x_i) = \frac{\sum_{j=1}^{n} w_j C_j(b_q, x_i)}{\sum_{j=1}^{n} w_j}.$$

Step 3 Calculate the partial discordance indices $D_j(x_i, b_q)$ and $D_j(b_q, x_i)$ for each criterion g_j. We compute the partial discordance index $D_j(x_i, b_q)$ for the criterion g_j according to the increasing direction of preference as follows (see Figure 6):

$$D_j(x_i, b_q) = \begin{cases} 0, \text{ if } g_j(b_q) - g_j(x_i) < p_j(b_q), \\ 1, \text{ if } g_j(b_q) - g_j(x_i) \geq v_j(b_q), \\ \frac{g_j(b_q) - g_j(x_i) - p_j(b_q)}{v_j(b_q) - p_j(b_q)}, \\ \text{ if } g_j(b_q) - v_j(b_q) < g_j(x_i) \leq g_j(b_q) - p_j(b_q). \end{cases}$$

The partial discordance index $D_j(b_q, x_i)$ as follows:

$$D_j(b_q, x_i) = \begin{cases} 0, \text{ if } g_j(x_i) - g_j(b_q) < p_j(x_i), \\ 1, \text{ if } g_j(x_i) - g_j(b_q) \geq v_j(x_i), \\ \frac{g_j(x_i) - g_j(b_q) - p_j(x_i)}{v_j(x_i) - p_j(x_i)}, \\ \text{ if } g_j(x_i) - v_j(x_i) < g_j(b_q) \leq g_j(x_i) - p_j(x_i). \end{cases}$$

Step 4 Calculate the outranking index $S(x_i, b_q)$ that shows outranking credibility of x_i over b_q assuming $S(x_i, b_q) \in [0, 1]$ as follows:

$$S(x_i, b_q) = C(x_i, b_q) \prod_{j=1}^{n} \frac{1 - D_j(x_i, b_q)}{1 - C_j(x_i, b_q)},$$

where $j = 1, \ldots, n$ and $D_j(x_i, b_q) > C_j(x_i, b_q)$.

Step 5 The DM defines the cutting level λ. Usually, the value of the cutting level is $\lambda \in [0.5, 1]$. This level defines the minimal value of outranking indices accepted for outranking of one alternative over the other one. The value of the outranking index $S(x_i, b_q)$ is compared to the cutting level λ. Based on such comparison, the preference situation between two alternatives is specified: they can be indifferent, one preferred to the other one or they may be incomparable:

- if $S(x_i, b_q) \geq \lambda$ and $S(b_q, x_i) \geq \lambda \Longrightarrow x_i I b_q$, then the alternatives x_i and b_q are indifferent;
- if $S(x_i, b_q) \geq \lambda$ and $S(b_q, x_i) < \lambda \Longrightarrow x_i P b_q$ or $x_i Q b_q$, then the alternative x_i is strongly or weakly preferred to the boundary alternative b_q;
- if $S(x_i, b_q) < \lambda$ and $S(b_q, x_i) \geq \lambda \Longrightarrow b_q P x_i$ or $b_q Q x_i$, then the boundary alternative b_q is strongly or weakly preferred to the alternative x_i;
- if $S(x_i, b_q) < \lambda$ and $S(b_q, x_i) < \lambda \Longrightarrow x_i J b_q$, then the alternatives x_i and b_q are incomparable.

Part II. We exploit the constructed outranking indices $S(x_i, b_q)$ for the classification of each alternative in the following way. The DM selects the assignment procedure: pessimistic or optimistic or both. Then, we compare the outranking indices for each pair of the alternative x_i to be classified and each boundary alternative b_q to the cutting level λ with regards to the assignment procedure as it is shown in Figure 7.

The pessimistic (or conjunctive) procedure assumes that we start to compare the alternative x_i to the lower bound b_{q-1} of the highest class l_q ($q = s, \ldots, 1$) and continue in a decreasing order until we find such a lower bound b_{q-1} that is outranked by the alternative x_i: $x_i S b_{q-1}$. For the estimation of the outranking relation $x_i S b_{q-1}$, the outranking index $S(x_i, b_{q-1})$ is calculated. Then, we calculate the outranking index between the alternative x_i to be classified and the upper bound b_q of the class l_q: $S(x_i, b_q)$. We assign the alternative x_i to the class l_q if $S(x_i, b_{q-1}) \geq \lambda$ and $S(x_i, b_q) < \lambda$.

The optimistic (or disjunctive) procedure, on the other hand, assumes that we begin to compare the alternative x_i to the upper bound b_q of the lowest class l_q ($q = 1, \ldots, s$) and proceed in an increasing order until we find such an upper bound b_q that has strict preferences over the alternative x_i: $b_q P x_i$. Then, we calculate the outranking index between the alternative x_i to be classified and the lower boundary alternative b_{q-1} of the same class l_q: $S(x_i, b_{q-1})$ we assign the alternative to this class l_q if $S(x_i, b_{q-1}) \geq \lambda$ and $S(x_i, b_q) < \lambda$.

Classification according to one of the described procedures is unambiguous. If an assignment is considered according to both of the procedures, an alternative can be assigned to different classes. For example, an alternative can be assigned in the pessimistic procedure to a lower class than in the optimistic one. This ambiguity must be resolved with the help of the DM or by changing the assignment procedure [135].

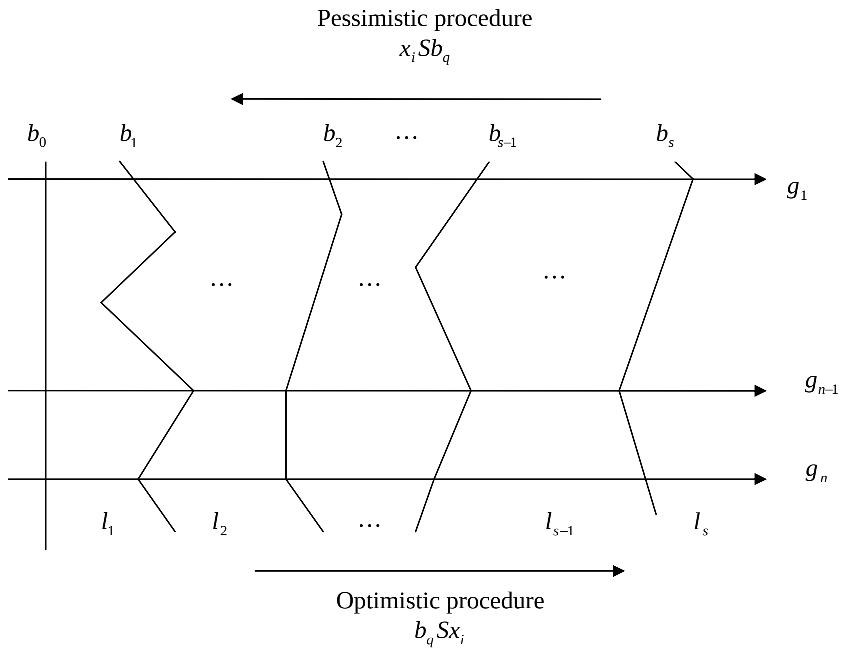

FIGURE 7 Assignment procedure of the ELECTRE TRI method

3.1.2 Extensions and Applications

The first outranking-based classification method, Trichotomic Segmentation, was introduced by Moscarola and Roy [111], [136]. Its big limitation is the inability to use more than three classes; nevertheless, as the first trial to classify within the framework of multicriteria decision theory made in 1977, it was a great step. Moreover, the method does work and its interactive modification has recently been used [74].

The main differences between Trichotomic Segmentation and ELECTRE TRI is the definition of the sets of upper and lower boundary alternatives as well as determination of several cutting levels in the former one. The definition of several cutting levels requires more cognitive effort from the DM when compared to ELECTRE TRI, where only one cutting level is defined. At the same time, the possibility of having several upper and several lower boundary alternatives allows the boundaries between classes be of a more versatile form. It also means that the upper bound of a class is not necessarily the lower bound for the class with a higher importance.

In [136], there is an example of an implementation of the Trichotomic Segmentation method for a small sized real-world problem with 2 upper boundary alternatives and 2 lower boundary alternatives, and 5 criteria with the range of 0-100 on each criterion. The modified Trichotomic Segmentation method, that is based on the continuous interaction with the DM, was improved for larger data sets with 600 alternatives estimated on 5 criteria with a range of possible values of 0-100 in [74].

The ELECTRE TRI method has successfully been applied to a large number of real-world problems. These include but are not limited to: environmental applications, such as estimation of impact on groundwater quality with sorting cropping systems [3], definition of national priorities when reducing greenhouse gas emissions [53], assessment of land-use stability [75], sorting of public transport zones for evolutive ticket pricing [117], software evaluation [128], and in education estimation of skills and qualification [145]. Thus, from this review of literature the method is able to solve problems with the number of alternatives up to 100, evaluated on up to 100 criteria with a range of values on each of them between 0-100, and classified in up to 5 classes.

The ELECTRE TRI model is sensitive to the variation of the parameters values. That is why the SMAA-TRI method has been developed for sensitivity and robustness analysis [151]. We consider SMAA-TRI in more detail in Section 3.4. There are other more traditional procedures that has been developed for studying imprecise parameters of ELECTRE TRI. In [106], a sensitivity analysis with extremes values for possible parameter values is undertaken. Dias and Mousseau [30] have developed the interactive software IRIS that performs a robustness analysis assuming all parameters of the ELECTRE TRI model to be defined imprecisely. The tool communicates with the user, asking to provide constraints for all the parameters and inform if some contradiction appears.

3.1.3 Concluding Remarks

In ELECTRE TRI, it is necessary to define a large number of parameters besides the boundary alternatives that separate classes: the thresholds and the weights for each criterion as well as the cutting level. The specification of such parameters directly may be cognitively difficult, although possible, for the DM. For instance, the definition of thresholds is not an easy task for the DM and requires from him or her an ability to understand the meaning of each threshold and to operate with differences on criteria values (that is not a typical cognitive operation for the DM). Moreover, even though using the thresholds has advantages when very conflicting criteria are involved, one can claim that they do not have a clearly defined physical or psychological interpretation [16]. Another evaluative aspect is the DM's time costs: the less information we ask from the DM, the less is the cost in time; however, more information allows creating more comprehensive models. Although exercising the parameter specification may help the DM to understand his or her preferences, such acquisition process requires from him or her a lot of cognitive effort. This is the reason for several studies in the MCDA theory devoted to the preference elicitation procedures. For outranking methods, DIVAPIME [113] and SRF [46] techniques have been developed for the purpose of weights elicitation.

Based on the preference disaggregation approach, principles (see Section 3.3) there have been models created for all the ELECTRE TRI parameters elicited at the same time [118]. However, such models appear to be quite complex. That is why several other mathematical models (that are also based on preference disaggregation approach) have been developed for eliciting parts of ELECTRE TRI parameters while assuming the rest of parameters to be known. These models have been used for eliciting weighs and cutting level [116], boundary alternatives [124], and veto thresholds [31]. At the same time, the ELECTRE TRI Assistant software [119], [124] has been developed for user-friendly parameter elicitation from assignment examples. We consider some preference elicitation methods in more detail in the next section.

3.2 ELECTRE TRI Preference Elicitation Methods

As was mentioned earlier, for the construction of the ELECTRE TRI model, the weights, the preference, the indifference and the veto thresholds for each criterion as well as the cutting level should be asked from the DM. Even though such parameters may be understandable for a DM in the situations with very conflicting criteria, their precise values may be difficult to define if asked directly. For example, the analyst can ask a question: "Please, specify the maximal value of the difference between two alternatives on the criterion g_j such that they remain indifferent for you?" Even professionals in the problem area may not be able to answer because they are not used to compare differences between criteria values. Such difficulties create preconditions for the development of indirect ELECTRE TRI parameter elicitation methods.

Researchers have been searching for ways to simplify the procedure of preference elicitation and found that DMs easily compare two alternatives and are able to select the better one when these alternatives differ only on some criteria. A pairwise comparison of such alternatives allows defining parameters of ELECTRE TRI in the DIVAPIME method. It is also convenient for a DM to give examples of classifications of alternatives that he or she is used doing in everyday practice. This property has been used in preference disaggregation approach presented in Section 3.3 and, later, it has been extended in ELECTRE TRI Assistant for the development of parameters of the ELECTRE TRI model. In the next two sections, we discuss DIVAPIME [113] briefly and ELECTRE TRI Assistant [119], [124] in more detail.

3.2.1 DIVAPIME

DIVAPIME (Determination d'Intervalles de Variation pour les Parametres d'Importance des Methodes ELECTRE) [113] is an Elicitation Technique for Importance Parameters (ETIP) for the ELECTRE family of methods. DIVAPIME is an indirect technique. This means that the values of the ELECTRE TRI parameters are not asked directly from the DM but obtained through a comparison of alternatives. During pairwise comparisons of alternatives the DM defines which one of two alternatives is preferred or states that they are indifferent.

For the ELECTRE TRI method, it is necessary to specify boundary alternatives, a cutting level and for each criterion weight and thresholds: preference, indifference and veto. It has been proposed, with DIVAPIME, to search for the weights and thresholds by solving a clustering problem in the space of ordered criteria. The DIVAPIME method assumes boundary alternatives and the cutting level to be defined by the DM.

Clustering the space of ordered criteria means grouping criteria with a similar importance into one group when compared to the criteria in the other groups. Groups of criteria are ordered, so dividing the space of weights can be done by finding the upper and lower bounds for each of them. In order not to mix the main problem of this work - classification of the set of alternatives - with the problem in this section of grouping weights, in what follows, we will refer to the latter problem as finding upper and lower bounds for the weights or an interval of weights. Here, an interval of weights is defined for a group of criteria that are of equal importance to the DM. It was proposed in [113] to define such intervals through the construction of a non-empty polyhedron in the space of weights. Clustering the weights with DIVAPIME assumes that the upper bound of a weight group is not necessarily the lower bound for the neighboring group of weights with a higher importance. Thus, for each group, upper and lower bounds should be defined.

The preference and indifference thresholds are calculated for the neutral (neither attractive nor unattractive) and the attractive (the most desirable - maximal as we consider maximization as the optimum) evaluations on each criterion. The veto threshold is determined in the same way as the preference and indifference thresholds, if necessary. Next, we discuss briefly the procedure for the elicitation of weights implemented by grouping the space of the admissible weights.

The algorithm of the DIVAPIME preference elicitation procedure

Step 1 Ranking the weights by the importance or pre-ordering criteria.

Step 2 Clustering the weights with a similar importance, that is finding groups in the weight space (collection of "similar" weights in the group that are "different" from the weights in the other groups).

Step 3 Evaluation of distance between groups of weights and definition of the lower bound for each group.

Step 4 Providing an upper bound for each group of weights.

Step 5 Reducing the polyhedron of weights and in such a way cutting the number of groups if it is possible.

At the end of the preference elicitation procedure, there are groups of criteria, each of which is of a similar importance for the DM. A group of weights has upper and lower bounds that define the range of the weights acceptable for the current criteria group.

The threshold acquisition is performed through a dialog with the DM. It could be difficult for the DM to estimate significant differences between two criterion values. In order to infer the values of these thresholds, the DM is asked to define the neutral and the attractive values for each criterion. The neutral value is neither attractive nor unattractive estimation, while the attractive value is the most desirable value on the criterion. The preference and the indifference thresholds for the neutral value of criterion are searched for on the interval between the neutral and the attractive values of this criterion. The search is organized by a dichotomy method. The minimal value of criterion indifferent to the neutral value is considered as an indifference threshold, and the maximal value of criterion that is preferred to the neutral value is defined as the preference threshold for the neutral value. In the same way, the thresholds for the attractive values are explored during dichotomic search on the interval between the neutral and the attractive values and by comparing the result of segmentation with the attractive value.

The preference and the indifference thresholds are searched for the neutral and for the attractive values. The neutral and the attractive values of a threshold bound the interval for all possible values of thresholds. Even though the ELECTRE TRI method requires one weight value (and similarly one value of each threshold) for each criterion, it is not specified in DIVAPIME how to select this particular weight (threshold) value from the interval.

The idea of finding the ELECTRE TRI parameters by a pairwise comparison of fictitious alternatives is promising; however, it is time-consuming with the DIVAPIME method and requires a lot of cognitive effort from the DM. Another approach, used in the ELECTRE TRI Assistant method, intends to overcome these drawbacks.

3.2.2 ELECTRE TRI Assistant

The ELECTRE TRI Assistant procedure [119] is based on the preference disaggregation approach ideas presented in Section 3.3, according to which for a DM it is easy to give examples of the classification of alternatives to classes that he or she is used doing in everyday practice. For example, the DM can say: "Usually, I classify the alternative x_i to the class l_q, the alternative x_r to the class l_k" and so on. Based on the examples described, there is a possibility of a holistic [118] or a partial [113], [116], [124] inference of parameters of the ELECTRE TRI classification model with mathematical programming techniques. The model developed should reassign the given set of classification examples as closely as possible with a minimal misclassification error. Partial inference of weights is done by assuming that other parameters (thresholds and boundary alternatives) are fixed. The other possibility is to infer the preference and the indifference thresholds by assuming the weights and boundaries to be constant. In what follows, the holistic preference elicitation procedure is considered.

In ELECTRE TRI Assistant, except for the assignment examples, the DM should define a cutting level. For simplicity, it is assumed that the outranking relation follows only a concordance rule without a necessity to take into account non-discordance. In this case, there is no need to define a veto threshold. The other parameters such as the preference and the indifference thresholds as well as boundary alternatives and weights of criteria are obtained by solving some optimization problems to be described below by means of a mathematical programming technique.

The model developed by ELECTRE TRI Assistant should be compatible with the assignment examples if the ELECTRE TRI method should be able to provide a classification of these examples in the same way as the DM does. The experimental studies [116] show that it is enough to have $2n$ (where n is the amount of criteria) assignment examples $X^* \in X$ (where X is the set of alternatives) for eliciting the weights if the other parameters are fixed. The results of this study were also used in the ELECTRE TRI Assistant method but enlarged for the holistic inference of all parameters including the preference and the indifference thresholds as well as the boundary alternatives. In order to develop the optimization problem, we have to define its variables, objective function, and constraints.

The algorithm of the ELECTRE TRI Assistant preference elicitation procedure

Step 1 Define variables that should be optimized. In the ELECTRE TRI method, the outranking index is used as a measure of indifference or preference between each alternative and each boundary alternative of each class. That is why it is necessary to search for the following variables: preference $p_j(\cdot)$ and the indifference $q_j(\cdot)$ thresholds and weight w_j for each criterion g_j $(j = 1, \ldots, n)$ as well as boundary alternatives b_q $(q = 0, \ldots, s)$.

The cutting level allows defining which values of the outranking index are large enough for an outranking relation to be true and which are not. As was defined in Section 3.1.1 for the assignment of the alternative $x_i \in X^*$ $(i = 1, \ldots, m^*)$ to the class l_q $(q = 0, \ldots, s)$ according to the pessimistic assignment procedure, two conditions should be satisfied: $S(x_i, b_{q-1}) \geq \lambda$ and $S(x_i, b_q) < \lambda$. That is why two slack variables, z_i and y_i, are introduced. These define the differences $S(x_i, b_{q-1}) - \lambda$ and $\lambda - S(x_i, b_q)$, respectively, as follows:

$$\begin{aligned} z_i &= S(x_i, b_{q-1}) - \lambda, \\ y_i &= \lambda - S(x_i, b_q), \end{aligned}$$

where $z_i \geq 0$ and $y_i \geq 0$.

Step 2 Define an objective function. The goal of the inferring procedure is to find a model that best matches the assignment examples offered by the DM and to minimize misclassification error (following the preference disaggregation approach ideas presented in Section 3.3). For simplicity, it is assumed that slack variables z_i and y_i are similar for each alternative $x_i \in X^*$ to be misclassified to any class. The authors of [119] assume that minimization of the slack between the outranking index and the cutting level increases the probability of an alternative to be assigned to a class correctly. That in turn would allow obtaining an optimal model that corresponds to the assignment examples proposed by the DM:

$$\max_{x_i \in X^*} \min\{z_i, y_i\},$$

where $z_i \geq 0$ and $y_i \geq 0$ are slack variables for the assignment examples, that are, alternatives $x_i \in X^*$.

In order to take into account any possible inaccuracy of the model, when compared to the DM's assignment examples, it is necessary to consider the overall ability to assign an alternative to a "correct" class:

$$\max_{x_i \in X^*}(\min\{z_i, y_i\} + \varepsilon \sum_{x_i \in X^*}(z_i + y_i)),$$

where $\varepsilon > 0$ is a small value; $\sum_{x_i \in X^*}(z_i + y_i)$ is a sum of slack variables for each assignment example $x_i \in X^*$.

The differentiable form of the same function is:

$$\begin{array}{l} \max_{x_i \in X^*}(\alpha + \varepsilon \sum_{x_i \in X^*}(z_i + y_i)), \\ \text{s. t. } \alpha \leq z_i, \alpha \leq y_i, \\ i = 1, \ldots, m^*, \end{array} \tag{1}$$

where m^* is the number of assignment examples in X^*.

Step 3 Define constraints for the problem. The following constraints for the slack variables should also be taken into account:

$$\begin{array}{l} S(x_i, b_q) - z_i = \lambda, \\ S(x_i, b_{q+1}) + y_i = \lambda, \\ \lambda \in [0,5;1], \end{array}$$

as well as the constraints for the thresholds and importance weights of criteria:

$$\begin{array}{l} g_j(b_{q+1}) \geq g_j(b_q) + p_j(b_q) + p_j(b_{q+1}), \\ p_j(b_q) \geq q_j(b_q), w_j \geq 0, q_j(b_q) \geq 0, \\ j = 1, \ldots, n; q = 1, \ldots, s, \end{array}$$

where b_q, w_j, $p_j(\cdot)$, $q_j(\cdot)$ are variables that are searched for each criterion g_j; and $p_j(\cdot)$ and $q_j(\cdot)$ depend on the alternative considered.

Step 4 Create an optimization problem and solve it by means of some mathematical programming technique. We get the problem including (1) and constraints needed in the form:

$$\begin{array}{l} \max_{x_i \in X^*}(\alpha + \varepsilon \sum_{x_i \in X^*}(z_i + y_i)), \\ \text{s.t. } \alpha \leq z_i, \alpha \leq y_i, \\ \frac{\sum_{j=1}^{n} c_j(x_i, b_q)}{\sum_{j=1}^{n} w_j} - z_i = \lambda, \\ \frac{\sum_{j=1}^{n} c_j(x_i, b_{q+1})}{\sum_{j=1}^{n} w_j} + y_i = \lambda, \\ \lambda \in [0,5;1], \\ g_j(b_{q+1}) \geq g_j(b_q) + p_j(b_q) + p_j(b_{q+1}), \\ p_j(b_q) \geq q_j(b_q), w_j \geq 0, q_j(b_q) \geq 0, \\ i = 1, \ldots, m^*; j = 1, \ldots, n; q = 1, \ldots, s. \end{array}$$

This is a nonlinear problem with $2m^* + 3ns + n + 2$ variables and $4m^* + 3ns + 2$ constraints, where m^* is the number of assignment examples; n is the number of criteria; and s is the number of classes. The quality of the model developed with the assignment examples is assessed by the overlap between the results of ELECTRE TRI supported by ELECTRE TRI Assistant and the DM's opinions about classification. If the model does not provide a satisfactory classification it is proposed for the DM to rethink the assignment examples [118].

3.2.3 Concluding Remarks

The two parameter elicitation methods described above use different approaches. While ELECTRE TRI Assistant allows holistic elicitation of DM's preferences in the form of assignment examples, DIVAPIME utilizes results of pairwise comparisons of alternatives. Each method has its advantages and drawbacks. The ELECTRE TRI Assistant method involves modest cognitive effort from the DM, while in DIVAPIME, the cognitive loading of the DM is larger.

On the other hand, ELECTRE TRI Assistant includes solving a nonlinear problem that requires computational time and a powerful computer. However, both of the indirect parameter elicitation methods described simplify the process of parameter acquisition when compared to obtaining them directly from the DM.

3.3 Preference Disaggregation Approach

The researchers were searching for ways to simplify the parameter elicitation in the form specified by each model. They considered the difficulties that the DM meet while expressing his or her preferences in a particular form of parameters defined by the model, and noticed that the DM may have no time or willingness to understand the meaning of the parameters of the model. On the other hand, for the DM it may be easier to provide the examples of ready decision that he or she used making in the past; or decisions made for realistic alternatives not considered in the set of given alternatives; or decisions for a limited set of alternatives from the set of given alternatives [72]. This fact was considered by developers of preference disaggregation approach (PDA) [72]. Hence, in PDA, it is not required for the DM to express his or her preferences in the form of some specific model parameters but to provide the analyst with ready decisions. Such information should be enough to develop the decision aiding model.

The relation between the criteria evaluations and the examples of decisions are analyzed with an ordinal regression-based technique. Then, an aggregation model is developed in such a way that it resolves, as closely as possible, the provided decisions. The UTA (UTilités Additives) method is the first PDA method developed for the ranking type of MCDA problems and was introduced by Jacquet-Lagreze and Siskos [71]. The first versions of the method assume a weighted sum preference model, for which parameters are estimated with linear programming. However, more complex models have been developed later. A summary of the development of PDA methodology in twenty years until the year 2001 may be found in [72]. Next, we consider the MCDA classification method developed in the framework of PDA.

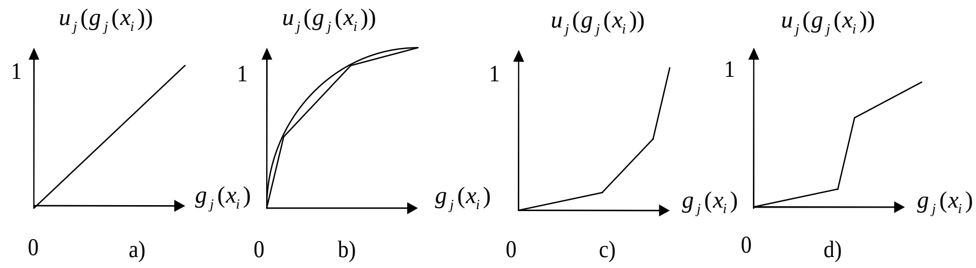

FIGURE 8 Approximated UTADIS criteria functions

3.3.1 UTADIS

First attempts to develop the classification models in the framework of PDA appear during 60s and have been based on the discriminant function [37]. The fist MCDA method, UTADIS, for ordinal classification has been developed based on the UTA method and discriminant analysis [29], [37]. The method assumes that the DM behaves according to the utility function. In particular, UTADIS is based on the additive weighted sum utility function. The method tries to reduce the cognitive loading of the DM and it does not require him or her to express preferences in the form of certain model parameters. Indeed, following the PDA approach, in UTADIS, the information about assignment examples for each class is elicited from the DM. Then, this information is used for the identification of the classification model parameters with a mathematical programming technique. The model developed should reassign the given set of classification examples as closely as possible with a minimal misclassification error.

Initially, the UTADIS method assumes that the set of assignment examples $X^* = \{x_1, \ldots, x_{m^*}\}$ is available, estimated on the set of criteria $G = \{g_1, \ldots, g_n\}$ that may have verbal or numerical values and be defined by linear or nonlinear criterion function (for instance, see Figure 8). For each discrete criterion g_j from the set G, the scale of values $S(g_j) = \{g_{j1}, \ldots, g_{jt}\}$ (where t is the number of values on the scale of criterion g_j) is defined. For the continuous type of criterion, the function is available and its minimal and maximal values $[g_j^{min}, g_j^{max}] \in S(g_j)$. The utilities of criteria functions $u_j(g_j(x_i))$ for $x_i \in X$ and $g_j \in G$ are supposed to be monotone functions (linear or nonlinear) defined on the scales of criteria. These functions allow different types of scales (both numerical and verbal) with different measures to be reduced to a new scale in the interval $[0, 1]$. We assume that bigger values are better and solve the maximization problem. Thus, all criteria should be reduced to increasing scales.

The form of the criteria utility functions may be different depending on the risk-attitude behavior of the DM and can be defined through regression analysis. Thus, for a risk-neutral behavior this function is linear (see Figure 8a). A risk-averse behavior with the inclination of the DM to be satisfied with the acceptable performance leads to a concave function type (see Figure 8b), while the tendency of the DM to search for the top performance and defined as risk-prone behavior

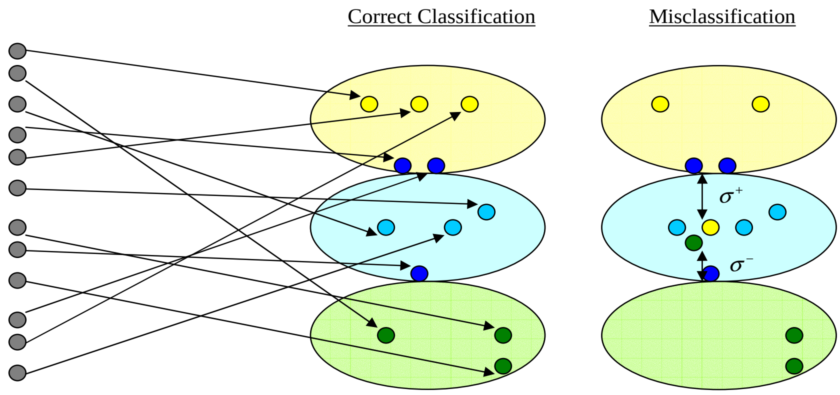

FIGURE 9 Correct classification and misclassification

can be presented with a convex function (see Figure 8c). Finally, mixed type of behavior is presented in Figure 8d.

The algorithm of the UTADIS method

Step 1 Select the type of aggregation function. The primary version of the UTADIS method uses an additive utility function for the construction of the criteria aggregation model:

$$u(x_i) = \sum_{j=1}^{n} w_j u_j(g_j(x_i)), \tag{2}$$

where $u(x_i)$ is the global utility of the alternative $x_i \in X^*$, and $u_j(g_j(x_i))$ is utility of the alternative x_i $(i = 1, \ldots, m^*)$ on the function of criterion g_j $(j = 1, \ldots, n)$ (called in [37] as marginal utility functions), furthermore, w_j is the weight of the criterion g_j such that the set of weights is nonnegative and normalized:

$$w_j \in W = \left\{ w_j \in R^n \mid w_j \geq 0 \text{ and} \sum_{j=1}^{n} w_j = 1 \right\}. \tag{3}$$

The criteria weights are parameters to be defined through a PDA in UTADIS.

Step 2 Approximate each nonlinear criterion function (for instance, see Figure 8b) with piecewise linear functions as follows. It is assumed that the nonlinear function of the criterion g_j taken values on some interval $[g_j^{min}, g_j^{max}] \in S(g_j)$ is divided into z linear subintervals $[g_j^0, g_j^1[, \ldots, [g_j^{z-1}, g_j^z]$ such that:

$$g_j^k = g_j^{min} + \frac{k}{z}(g_j^{max} - g_j^{min}),$$

where $k = 0, \ldots, z$ and $g_j \in G$.

The utility of each subinterval of the piecewise linear function is restricted by the utilities of the breakpoints $u_j(g_j^0) \leq \cdots \leq u_j(g_j^z)$. Then, the utility $u_j(g_j(x_i))$

of the alternative x_i on the criterion g_j that belongs to the subinterval $[g_j^k, g_j^{k+1}[$ is defined:

$$u_j(g_j(x_i)) = u_j(g_j^k) + \frac{g_j(x_i) - g_j^k}{g_j^{k+1} - g_j^k}(u_j(g_j^{k+1}) - u_j(g_j^k)),$$

where $k = 0, \ldots, z$ and $g_j \in G$.

From this condition, we can see that in order to classify an alternative, it is necessary to know all the utilities of the breakpoints $u_j(g_j^0) \leq \cdots \leq u_j(g_j^z)$ for all subintervals $[g_j^k, g_j^{k+1}[$ $(k = 0, \ldots, z)$ on all criteria $g_j \in G$. Thus, this set of utilities of breakpoints is a set to be defined through a PDA in UTADIS.

Step 3 Define classes by classification rules. The global performance of the alternative is estimated by aggregating the performances of the alternative on each criterion and taking into account the criteria weights with regards to equations (2) and (3). Then, each alternative should be classified by comparison of its global utility to the utility thresholds of the classes according to the following rule: the alternative is assigned into the class l_q $(q = 1, \ldots, s)$ if its utility is bigger than the lower utility threshold of the class and is smaller than the upper threshold of the class:

$$\begin{cases} \text{if } 0 \leq u(x_i) < u_1, \text{ then } x_i \in l_1, \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ \text{if } u_{q-1} \leq u(x_i) < u_q, \text{ then } x_i \in l_q, \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ \text{if } u_{s-1} \leq u(x_i) \leq 1, \text{ then } x_i \in l_s, \end{cases} \tag{4}$$

where the set $\{u_1, \ldots, u_{s-1}\}$ defines utility thresholds between neighboring ordered classes $\{l_1, \ldots, l_s\}$, respectively. In order to be able to perform the classification according to this set of rules, it is necessary to know the utility thresholds between all classes. This set of parameters will be defined through PDA in UTADIS.

Step 4 Define the objective function. We consider the original version of the UTADIS method [37], for which the goal is to develop a classification model that reassigns the set of assignment examples $x_i \in X^*$ as closely as possible to the given ordered classes. In such a way, the misclassification error should be minimized. For instance, in Figure 9 (on the right), two alternatives yellow and green that should be in the classes with yellow and green alternatives, respectively, appear in the class with blue alternatives. Then, for correcting such misclassifications it is necessary to increase the utility of the yellow alternative and decrease the utility of the green alternative.

Step 5 Define variables that should be optimized. For this purpose, two slack variables σ_i^+ and σ_i^- are introduced. In case of the slack variable σ_i^+, in order to classify the alternative x_i correctly into the class l_q, it is necessary to increase its global utility by $u_q - u(x_i)$; while the slack variable σ_i^- defines misclassification of the alternative x_i such that in order to assign it correctly into the class l_q its global utility should be decreased by $u(x_i) - u_{q-1}$. By including slack variables into the classification rules, the condition (4) changes in the following way:

$$\begin{cases} \text{if } u(x_i) - \sigma_i^- < u_1, \text{ then } x_i \in l_1, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \left.\begin{array}{l} \text{if } u(x_i) - \sigma_i^- < u_{q-1}, \\ \text{if } u(x_i) + \sigma_i^+ \geq u_q, \end{array}\right\} \text{ then } x_i \in l_q, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \text{if } u(x_i) + \sigma_i^+ \geq u_{s-1}, \text{ then } x_i \in l_s. \end{cases} \tag{5}$$

Thus, the variables of the model to be estimated are utility thresholds between all classes, utilities of the breakpoints of criteria functions, criteria weights, and slack variables.

Step 6 Define constraints for the problem. In addition to the constraints defined by the classification rules (5), the constraints for the weights to be non-negative and normalized (3) should be considered.

Step 7 Create an optimization problem and solve it by means of some mathematical programming technique. Taking into account the goal of the original UTADIS method, minimization of classifications errors of the assignment examples $x_i \in X^*$ provided by the DM, the following mathematical programming model has been developed for the estimation of the above-mentioned parameters:

$$\begin{array}{l}
\min \left\{ \sum_{q=1}^{s} \left[\sum_{x_i \in l_q} \frac{(\sigma_i^+ + \sigma_i^-)}{m_q} \right] \right\} \\
\text{subject to} \\
u(x_i) - u_1 - \sigma_i^- < \epsilon, \text{ for } x_i \in l_1, \\
\left.\begin{array}{l} u(x_i) - u_{q-1} - \sigma_i^- < \epsilon, \\ u(x_i) - u_q + \sigma_i^+ \geq \epsilon, \end{array}\right\} \text{ for } x_i \in l_q; (2 \leq q \leq s-1), \\
u(x_i) - u_{s-1} + \sigma_i^+ \geq \epsilon, \text{ for } x_i \in l_s, \\
\sigma_i^+ \geq 0; \sigma_i^- \geq 0, \\
u_q - u_{q-1} \geq \epsilon, \\
u(x_i) = \sum_{j=1}^{n} w_j u_j(g_j(x_i)), \\
w_j \geq 0, \ \sum_{j=1}^{n} w_j = 1, \\
u_j(g_j(x_i)) \text{ increasing function}, \\
u_i(g_j^{min}(x_i)) = 0; u_i(g_j^{max}(x_i)) = 1, \\
\left.\begin{array}{l} u_j(g_j(x_i)) = u_j(g_j^k) + \frac{g_j(x_i) - g_j^k}{g_j^{k+1} - g_j^k}(u_j(g_j^{k+1}) - u_j(g_j^k)), \\ u_j(g_j^{k+1}) - u_j(g_j^k) \geq 0, \end{array}\right\} \begin{array}{l} \text{if } u_j(g_j(x_i)) \\ \text{is nonlinear,} \end{array} \\
\text{for all } x_i \in X^*,
\end{array}$$

where σ_i^- and σ_i^+ are two classification errors for the alternative x_i when misclassified in upper and lower classes, respectively, when compared to the correct assignment into the class l_q; m_q is the number of assignment examples $x_i \in X^*$ classified to the class l_q; $u(x_i)$ is the global utility of the alternative x_i and $u_j(g_j(x_i))$ is the utility of the alternative x_i on the criterion g_j; u_q is the utility threshold between classes l_q and l_{q+1}; w_j is the weight of the criterion g_j; $[g_j^k, g_j^{k+1}[$ with

($k = 0, \ldots, z$) is the subinterval of the piecewise linear function on the criterion g_j; ϵ is a small positive constant introduced for avoiding situations where $u(x_i) = u_q$ (for details, see [37]).

The constraints of this mathematical programming model have a nonlinear form and solving this nonlinear programming problem could be challenging. In [37], two heuristics have been introduced in order to overcome this problem. The first one assumes that nonlinear criterion functions are divided in subintervals in such a way that there is at least one assignment example belonging to each subinterval. Another heuristics takes into account the distribution of assignment examples from different classes on each criterion's scale. If the first heuristics simplifies computational efforts when developing the classification model, the second one increases the stability of the additive utility classification models developed. Let us next discuss the modifications to the basic version of UTADIS and practical applications of the method in different problem areas.

3.3.2 Extensions and Applications

As has been mentioned above, the first version of UTADIS [29], [37] was developed assuming an additive utility function of the DM's preferences when solving the classification problem by minimizing misclassification error of the set of assignment examples provided by the DM. Later, several versions of UTADIS (UTADIS I, II, III) have been developed [36], [171]. In UTADIS I, at the same time with the original goal (minimization of misclassification errors) the additional goal is optimized: it maximizes the distances between correctly classified alternatives and utility thresholds. For modeling purposes, UTADIS II applies mixed-integer programming instead of linear programming, and minimizes the total number of misclassification errors instead of minimizing their magnitude. Finally, UTADIS III combines UTADIS I and UTADIS II modifications.

Another method developed in the framework of PDA is the MHDIS method (Multi-group Hierarchical DIScrimination) [35], [37]. This method has been developed considering interactive sequential classification. At each step of MHDIS, the decision about the assignment of not yet classified alternatives into the most preferred class (from not yet considered ones) is taken. In such a way, the alternatives classified in the previous steps as well as considered classes are not examined in the following iterations. Another distinction of the MHDIS method when compared to the UTADIS method are objectives taken into account. In addition to minimization of the misclassification errors, maximization of the clarity of the performed classification and minimization of the number of misclassifications (in a way similar to UTADIS II and III) are considered.

In several works, the properties of the UTADIS family of methods have been studied. Zopounidis and Doumpos examine UTADIS methods and compare their performance to well-known statistical techniques, such as discriminant analysis and logit and probit analysis [169]; and in [37] a comparison of UTADIS to several MCDA methods, such as rough sets and ELECTRE TRI is done. In [38], the authors study technical parameters of the UTADIS model, such as ϵ, and the ways for dividing criteria functions into subintervals. Mousseau and colleagues [114] have investigated the possibility of including the size (absolute or relative) of each class into the UTADIS method.

The UTADIS method has been implemented in the following multicriteria decision support systems: PREFDIS (PREFerences DIScrimination) [170] and FINCLAS (FINancial CLASsification) [171]. PREFDIS provides users with an interface for modeling ordinal classification models with all four versions of the UTADIS method. FINCLAS, in addition to possibilities of developing classification models with all four versions of the UTADIS method, provides tools for modeling of financial problems that include but are not limited to a data base of financial statements and quantitative financial information, tools for credit analysis and financial forecasting, and graphical tools.

Most of the applications of the UTADIS methods deal with financial classification [36], [169], [171]. In particular, the methods have been applied for the prediction of companies business failures [169], for country risk assessment when estimating the creditworthiness of countries [170], and for estimation of companies' credibility for assisting banks in granting credits [171].

3.3.3 Concluding Remarks

The classification with the set of assignment examples is easy for the DM. He or she should not need a lot of cognitive effort for providing a decision that he or she is used doing in the past or for being able to make classification of some subset of given alternatives or realistic alternatives not considered in the set of given alternatives. It is usually easier to provide such decisions than to provide upper and lower boundary alternatives for each class.

The additive utility function that is used throughout the UTADIS methodology is straightforward and understandable. On the other hand, it has been criticized for not taking into account the interaction between criteria [127]. However, the attempts to develop multiplicative models that would overcome such weakness in the framework of PDA lead to computationally complex nonlinear models.

Another issue discussed in [37] is post-optimality analysis. According to it, the classification model developed with PDA and constructed based on the set of assignment examples may not be robust enough if the set of assignment examples changes, if for instance, the DM adds new assignment examples. That is why some additional checking based on the knowledge of the preferential system of the DM would be of help.

3.4 Stochastic Multicriteria Acceptability Analysis

The traditional way of representing preferences by means of exact parameter values of a specific model have drawbacks, and these values may not be always available. The main difficulties with extracting initial information in the form expected by the specific model may appear when the parameters of the model need to be accepted by several DMs; or when there are no exact values but just intervals of parameter values; or when there is some distribution of parameter values; or the information about parameters is absent. The last mentioned case is the most unlikely of these, but it is possible in situations where the values of parameters may change over time, or are to be obtained later on. Thus, obtaining exact information about parameter values may be difficult not only for multiple DMs but even for one DM. The other problems are discussed in Sections 2.2.1 and 5.1.9, and may be connected to the interpretation of preference information like weights in different models as well as behavioral aspects of humans when providing such information. Moreover, such parameters (e.g., weights) elicited for the same problem with different techniques may be different. One possible way to tackle with such difficulties is used in PDA [37], [72], where the DM is not required to define the preferences in the form of some specific model parameters. On the contrary, PDA methods work in a backward manner and search for such parameters of the model that most consistently resolve the examples of earlier decisions provided by the DM. In a similar way, the so-called preference information free methods proceed. Such methods have been considered in [6], [24], [42], and [133]. Recently, another method based on similar ideas appears [66].

The SMAA method [83], [85]-[87] is also a preference information free and inverse method developed based on the overall compromise criterion method created by Bana e Costa [6]. SMAA [83] was proposed as a multicriteria decision support system for multiple DMs in the situation of a complete absence of preference information (for example, criteria weights) or in a situation of incompleteness of preference information (for example, intervals or distribution of criteria weights or criteria values). Based on the inverse principle, it describes preferences that correspond to each possible decision or scenario. SMAA allows not only to perform choice, ranking or classification but explores the parameter space based on the Monte-Carlo simulation and finds parameters that support each particular scenario (selection of the alternative x_i to be the best one, or to be located at a particular rank, or to be assigned to a particular class). For instance, for the classification problem it finds acceptabilities for each alternative to be assigned to each class. It may also be used as a sensitivity and robustness analysis tool for parameters of the selected MCDA model, in a similar way as it is done for the ELECTRE TRI in SMAA-TRI [151].

As mentioned before, SMAA allows to take into account preferences of several DMs and finds some compromise parameters that would represent preferences of an averaged or typical DM of this group of DMs. Then, if some preferences of several DMs are available, they may be introduced in the method with intervals or distributions of parameters that are considered in the Monte-Carlo simulation. In this work, we refer to modeling of preferences of a single DM as imprecise or uncertain, not forgetting that the preferences of several DMs may be modeled in the same way.

The nature of Monte-Carlo methods is such that random parameter values for the model are simulated. Then, the results of the modeling corresponding to the same scenario are aggregated. The frequency of each scenario shows the probability of its appearance, while averaged parameter values correspond to the preferences of a typical DM when such a scenario is chosen. In Monte-Carlo simulation, the error margins are small due to the large number of iterations; thus, they do not have to be taken into account. Most of the existing SMAA methods are for the choice and ranking types of MCDA problems. For the detailed survey of SMAA methods developed so far, we refer to [153].

In SMAA methods developed so far, the information about criteria values and/or DM's preferences are assumed to be imprecise or uncertain. Thus, the alternative x_i is denoted as $\xi_i \in \chi$ when estimated with stochastic values $g_j(\xi_i)$ on each criterion g_j. The stochastic criteria evaluations are presented by a joint probability distribution $f(\xi_i)$ in the space $R^{m\times n}$. It is assumed that the DM can express his or her preferences in the form of weights of criteria. For different methods, the meaning of the weights may be different. That is why the analyst must explain to the DM the meaning of weights depending on the model used. We discuss and compare weights in different models later in Section 5.1.9. For the SMAA method, it is also important to specify the meaning of weights. For instance, when using the original SMAA method [83] that is based on additive value function, where the weights represent trade-offs between criteria. Within ELECTRE TRI, and, consequently, in SMAA-TRI, weight represents the number of votes supporting each criterion.

The weight $w \in W$ is assumed to be stochastic. Thus, depending on the model used for each scenario the set of favorable weight vectors $W(x_i)$ is defined. For instance, in the original SMAA method [83], the set of favorable weights maximizes the values of the additive value function and with such set of weights the alternative is considered to be the best one:

$$W(x_i) = \{w \in W : u(x_i, w) \geq u(x_r, w), r = 1, \ldots, m\},$$

and any weight from the set of favorable ones makes the overall utility of the alternative x_i to be better or equal when compared to any alternative from the given set.

The SMAA method provides the following indices: the acceptability index, the central weight vector, and the confidence factor. The *acceptability index* shows the variety of different preferences that support the alternative to be the best one. It is computed as a multidimensional integral over the criteria distributions $f_\chi(\xi_i)$ and the favorable weight space $f_W(w)$ as follows:

$$a_i = \int_{\xi_i \in \chi} f_\chi(\xi_i) \int_{w \in W(\xi_i)} f_W(w) dw d\xi.$$

The acceptability index divides the set of given alternatives into efficient ones (with $a_i \gg 0$) and inefficient ones (with $a_i = 0$ or near zero). The acceptability index is affected by the scaling of criteria. That is why scaling cannot be done arbitrary. For more details on the possible types of scaling for SMAA, see [87].

The *central weight vector* defines the averaged favoring weights with which the alternative is considered to be the best one. It is computed as a multidimensional integral over criteria and weights distributions $f_{\chi}(\xi_i)$ and $f_W(w)$, respectively:

$$w_i^c = \int_{\xi_i \in \chi} f_{\chi}(\xi_i) \int_{w \in W(\xi_i)} f_W(w) w dw d\xi / a_i.$$

In case several DMs taking part in the decision process, the central weight vector represents the preferences of a typical DM supporting the alternative. Thus, even if the preferences of a particular DM are different from the ones represented by the central weight vector, he or she can learn about the preferences that show the averaging preferences of the DMs that participated in the decision process. On the other hand, if only one DM participates in the decision aiding process, but he or she has uncertain preferences, then the central weight vector represents the averaged preferences of this DM.

The *confidence factor* estimates the level of accuracy for the alternative to be the best one. It is calculated as a multidimensional integral over the criteria distributions $f_{\chi}(\xi_i)$ as follows:

$$p_i^c = \int_{\xi_i \in \chi : u(\xi_i, w_i^c) \geq u(\xi_r, w_i^c)} f_{\chi}(\xi_i) d\xi.$$

The confidence factor evaluates the accuracy of the criteria values for detecting efficient alternatives. Thus, in the case of selection of the set of most efficient alternatives among a large set of given ones, the alternatives with a low confidence factor should not be chosen.

The original SMAA method has been designed only for the choice type of MCDA problems with one or several best alternative(s) to be selected. SMAA-2 [87], on the other hand, allows ranking of the set of given alternatives. The analysis of each alternative is enlarged holistically for all ranks. It also assumes not only additive but a general value function, in order to include additional preference information in a different form. Thus, new descriptive indices are introduced: the rank acceptability index, three k-best rank-type measures, and the holistic acceptability index. In SMAA-2, again the sets of nonnegative and normalized weights (see equation (3)) are simulated. From all simulated weight vectors the favorable once are selected in $W^r(\xi_i)$ with which the alternative ξ_i is successfully assigned into the rank r:

$$W^r(\xi_i) = \{w \in W : rank(\xi_i, w) = r\}.$$

Such weights should assign alternative to a particular rank according to the function:

$$rank(\xi_i, w) = 1 + \sum_{r=1}^{m} \rho(u(\xi_i, w) \geq u(\xi_r, w)),$$

where $\rho(true) = 1$ and $\rho(false) = 0$.

We are not going into more detail that can be found in [153], for example. Next, we introduce the SMAA-TRI method developed for the sensitivity and robustness analysis of the ELECTRE TRI parameters.

3.4.1 SMAA-TRI

In this work, we are mainly interested in the classification methods. In the framework of SMAA, the SMAA-TRI method has been developed as an extension of the ELECTRE TRI method. SMAA-TRI analyzes the stability of ELECTRE TRI model parameters arbitrarily distributed in the finite space in order to find such of them that support assignment of alternative to each class. SMAA-TRI allows not only classification, but sensitivity and robustness analysis of the parameter values of the ELECTRE TRI method [151]. As has been mentioned in Section 3.1.1, the ELECTRE TRI method is sensitive to the variation of the parameter values. That is why such analysis is important. Next, we consider SMAA-TRI in more detail.

The SMAA-TRI assumes that the parameters of the ELECTRE TRI model are specified in a different way for classification with SMAA-TRI. In particular, in [151], it is proposed to analyze the stability of the sets of boundary alternatives $B = \{b_0, \dots, b_s\}$, of weights $W = \{w_1, \dots, w_n\}$, and of criteria $G = \{g_1, \dots, g_n\}$ and cutting level λ. They are presented in a non-deterministic way with the following stochastic variables:

- Each boundary alternative b_q of the class l_q is presented with the stochastic variable $\phi_j(b_q)$ for the value of the alternative b_q on the criterion g_j. Because, in ELECTRE TRI, we consider classes to be ordered, the boundary alternatives between classes are ordered as follows:

$$b_s \mathrm{~P~} b_{s-1} \mathrm{~P~} \cdots \mathrm{~P~} b_0.$$

  The boundary alternatives between classes are assumed to be independently distributed and their distributions do not overlap. Then, all possible criteria values for the set of boundary alternatives B may be presented with the joint density function $f_\Phi(\phi)$ in the space $\Phi \subseteq R^{s-1 \times n}$.

- The weights W for criteria G are presented by joint density function $f_W(w)$ in the space of feasible weights W. Total lack of preference information is presented, in a 'Bayesian' spirit, by a uniform distribution of parameter values. For instance, such a representation for the weights in W is defined as $f(w) = 1/vol(W)$. However, if some information about distribution or about the intervals of weights is available, it can be included here. The weights are assumed to be non-negative and normalized (see formula (3)). This is the main difference with the ELECTRE TRI weights, for which there is no need for normalization.

- The cutting level λ is defined as a stochastic variable Λ with density function $f(\Lambda)$ in the interval $[0.5, 1]$.

To simplify the analysis, the rest of the parameters are assumed to be deterministic and fixed, even though they all can be modeled in a similar way. Thus, thresholds (indifference $q(\cdot)$, preference $p(\cdot)$ and veto $v(\cdot)$) are assumed to be deterministic and defined by the DM, and criteria values $g_j(x_i)$ $(i = 1, \dots, m,\ j = 1, \dots, n)$

of alternatives from X are supposed to be available. We define the set of all the deterministic values by $T = \{g_j(x_i), q_j(\cdot), p_j(\cdot), v_j(\cdot)\}$ for $(i = 1, \ldots, m, j = 1, \ldots, n)$. The SMAA-TRI method may be used in an iterative manner allowing the ELECTRE TRI parameters values to be updated at each iteration.

As with other SMAA methods, in the output of the SMAA-TRI method there is descriptive information in the form of acceptability indices for each alternative to be assigned into each class. If we define the classification function $class(\Lambda, \phi, w, T)$ based on the ELECTRE TRI method as follows:

$$x_i \in l_q \Longrightarrow q = class(\Lambda, \phi, w, T), \tag{6}$$

and the class membership function as follows:

$$d(x_i, l_q) = \begin{cases} 1, \text{if } class(\Lambda, \phi, w, T) = q, \\ 0, \text{otherwise.} \end{cases}$$

Then, the *acceptability index* a_i^q describes the share of possible parameter values that supports the assignment of the alternative x_i to the class l_q and is defined as a multidimensional integral over the finite parameter space as follows:

$$a_i^q = \int_{0.5}^{1} f(\Lambda) \int_{\phi \in \Phi} f_X(f_\Phi(\phi)) \int_{w \in W} f_W(w) d(x_i, l_q) dw d\phi d\Lambda. \tag{7}$$

For a stable and robust assignment, the class acceptability index is equal to 1 only for one class and 0 for the rest of classes; otherwise, the classification is not stable and robust with respect to the imprecise parameters.

The algorithm of the SMAA-TRI method

Step 1 Simulate all the parameters that are assumed to be stochastic according to uniform distribution, if not specified otherwise.

Step 2 Apply ELECTRE TRI for the classification of the set of initial data with the stochastic (simulated) and deterministic (fixed) parameters. Save all the results necessary for calculating SMAA indices.

Step 3 Compute the descriptive information in the form of SMAA-TRI acceptability indices according to (7) for each alternative x_i to be assigned to each class l_q.

Step 4 Provide the DM with the SMAA indices. If the DM is not satisfied with the obtained classification, suggest him or her to rethink the initial information (deterministic or stochastic) and specify it. Otherwise, if the DM agrees with the results of the classification the procedure is finished.

When compared to the more traditional approaches for sensitive analysis [106] and robustness analysis [30], SMAA-TRI has following advantages:

- The cognitive effort for defining the extremes of parameter values is reduced.
- With acceptability indices quantitative information about assignment to different classed is provided when compared to the range of possible classes in the more traditional methods.

- Parameters that are imprecise by nature, such as weights, may be more meaningful when elicited as imprecise values (when compared to exact parameter values elicitation in the more traditional methods).

Later in this work (see Section 4.1), we present a new SMAA-Classification method that extends the SMAA methodology for classification problems.

3.4.2 Extensions and Applications

Except the main direction of the SMAA methodology, represented by the SMAA and SMAA-2 methods, several other SMAA methods have been developed. The SMAA-O method [85] expands the SMAA-2 in order to process ordinal and cardinal criteria values at the same time. The ordinal values on the scales are mapped with ranks. It is assumed that the DM(s) can rank the alternatives on each criterion before the analysis starts. Alternatives that are considered to be equally good are placed on the same rank. The interval on the scales of the ordinal criteria does not contain any information and may be different for the same criterion. There is also a method developed for the situation where the utility function cannot be applied but where a pseudo-criteria should be used. For such cases, the SMAA-3 method [88] has been developed taking into account the experience of the ELECTRE family of methods and based on Electre III (see [138], [158]). There have also been techniques for eliciting and aggregating preferences based on belief functions [155]. An alternative way of modeling preferences when compared to weights are reference points. By defining reference points the DM specifies the goals and desired values for each criterion. There are two methods that have been developed for the modeling of reference points with SMAA: Ref-SMAA [86] and SMAA-P [90]. Ref-SMAA [86] uses reference points to model desired goal preferences of multiple DMs. Achievement scalarizing functions of different types may be used for characterizing non-dominated solutions. The SMAA-P method [90] is based on the prospect theory [76] that assumes the behavior of the DM to be risk-dependent (see Section 3.3.1).

In addition, research has been done for situations where the dependencies between criteria should be taken into account [84]. There has been another investigation for improving discrimination in large sets of given alternatives to be analyzed with SMAA by means of introducing cross confidence factor [89]. Moreover, extensive study has been devoted to the selection of the number of necessary and sufficient simulations in order to have an accurate computation with the Monte-Carlo technique used in SMAA [153]. The accuracy and complexity of the SMAA algorithms are discussed in the works of Tervonen et al. [152], [153].

The applications of different SMAA methods include infrastructure planning for the development of the harbor area in Helsinki, Finland [65], forest planning in Finnish Lapland [78], [79], and elevator planning [154].

3.5 Verbal Decision Analysis

The verbal decision analysis (VDA) methods have been developed by Larichev and his colleagues [100, 95], in order to aid in the decision making of unstructured problems. According to Newell, Shaw and Simon [123], it is difficult, in unstructured problems, to provide numerical evaluations on criteria as well as estimate dependencies between them. Usually, in unstructured problems, a DM faces a new problem or a problem that has new features when compared to the ones he or she has been solving and where information may be missing at the moment of decision making. The alternatives can have verbal or imprecise numerical estimations on criteria values. The absolute definition of scales of criteria can be absent and only a relative definition is at the disposal of the DM. For instance, it may be known that one value is better than another one, but the information about how much better it is may be absent. In such problems, criteria, instead of having objective scales, may only have subjective scales that are based on the DM's preferences, his or her knowledge, experience and intuition.

There can be uncertainties in the definition of the classes: in their number and boundaries, not forgetting the clustering problem. However, here we mainly deal with classification tasks where classes are defined (at least their number), and look at the uncertainty in the criteria and alternatives.

The idea behind the VDA methods is to take into account the cognitive sides of human behavior. These methods assume elicitation of DM's preferences in a natural language that sometimes can be the only way for the DM to express his or her thoughts. On the other hand, if numerical information is available, it is utilized. The elicitation of DM's preferences is adapted to the human "capacity limit", according to which human short term memory has limited abilities for storing information. When large volumes of information are to be processed simultaneously, a human usually applies different simplifications such as aggregation or division. He or she also tries to minimize efforts when making a decision in complex problems and uses simple strategies: easy cognitive operations such as addition and subtraction with a small number of variables [100]. Thus, increasing the number of criteria and the number of values on scales of criteria leads to decreasing the quality of decisions obtained by the DM.

Features of unstructured problems define the requirements for VDA methods. At the same time, it is necessary to take into account the fact that the logical rules of decision making used in the VDA methods should be mathematically valid for operating with verbal information. Even though, according to psychological experiments [92], it is natural for the DM to use simple mathematical operations such as addition and subtraction, they cannot be used when verbal values are estimated. Thus, we should search for other ways when developing rules for operating with verbal judgements. Another requirement is connected with the human factor in the decision aiding process: the DM's estimations are not necessarily accurate. Obviously, people can make errors and be inconsistent with their preferences. The error propagation leads to incorrect results. That is why we should be able to check the consistency of information obtained from the DM with his or her previous answers at each iteration of the decision aiding process. In spite of the DM's continuous involvement in the process of decision aiding, obtaining some final solution by the method may not be evident for him of her. Thus, the method should provide an explanation for the final result.

The main feature of the VDA methods is the ability to operate with qualitative information without direct transformation to a quantitative form. Usually, MCDA methods are not adapted for working with verbal information in a pure form: some converting rules are applied and the final result is obtained in numerical form or the DM is forced to express his or her preferences in a numerical form. However, there are cases, when it is difficult to give precise numerical estimation on criteria or the information can be lost during the transformation.

When developing VDA methods, researchers were searching for ways to avoid transformation of verbal criteria values into numerical analogies by the DM or by the method at the final output. They utilized the assumption that values on the scales of criteria are ordered according to their preference to the DM in such a way that the most desirable value has the maximal rank. Judging desirability is possible due to the nature of criteria. Thus, for each value on the scale of a criterion a rank corresponds. It is assumed that the difference between two neighboring values on the scales of criteria is the same for all pairs of neighboring values. When operating with ranks, it is possible to save the transparency of verbal values to the DM and allow him or her to make an evaluation of the alternatives with verbal values, while using ranks of verbal values on criteria scales inside the methods.

For different types of MCDA problems, Larichev and his colleagues developed several methods based on the principles of VDA [92], [95], [96], [99], [100]. Closed Procedures near Reference Situation (that is the abbreviation of Russian words: ZAPROS-LM) [100] allows partial rank-ordering of alternatives according to their importance for the DM, PAired COMpensation (PACOM) [100] selects the best alternative from the set of given alternatives, and ORdinal CLASSification (ORCLASS) [100], Subset Alternative Classification (SAC) [95] and CYCLE [93] methods classify a set of alternatives into ordered classes.

In general, any VDA method contains the following routines:

1. Interactive dialog between the DM and the algorithm, in the framework of which the DM has to classify some alternatives, but not all.

2. Based on the partial preferences inferred in the previous routine, decision aiding rules are developed with regards to the psychologically proved approaches to the decision making that correspond to the human processing abilities.

3. Information received from the DM is checked on the consistency with previously obtained information. In case that contradictions appear, the DM is asked to resolve them by changing one or both of the contradicting answers.

4. Explanation procedure for decisions obtained is at the disposal of the DM at the end of the method.

The classification with VDA is interactive. A preference elicitation procedure is built in the methods and is organized through a dialog with the DM: at each iteration, the DM assigns one alternative specifically selected by the VDA method to some class. Based on the answer of the DM, some alternatives may be automatically classified. The procedure is repeated until each alternative is assigned to exactly one class. The consistency of the information received from the DM is checked with the previously obtained one. In case of a contradiction, the DM is asked to rethink his or her answers. At the end of the classification, when each alternative is classified, the set of rules that describes classes can be obtained. Based on this set of rules, the DM can get an explanation about the appearance of each alternative in the class. According to the requirements of unstructured problems and taking into account features of classification tasks, the ORCLASS and the SAC methods have been developed. In the next section, the ORCLASS method will be discussed as well as its limitations, and after that, the SAC method is described to show how to overcome some of these limitations.

3.5.1 ORCLASS and SAC

The ORCLASS method [100] is the first of the VDA methods developed for classification. It allows assigning alternatives described with qualitative or numerical values into ordered classes. Thus, the initial information for the ORCLASS method is the following: the number of classes and the set of criteria with the scale of values for each of them. The values on the scales of criteria are ordered by importance: there is a rank for each criterion value on the scale, and the most desirable value has the maximal rank. The classes are also ordered in such a way that the most desirable class has the maximal rank. The set of alternatives involved in classification is obtained as a combination of values on the scales of criteria in a form of a so-called Cartesian product of the scales of criteria. Typically, alternatives with values on criteria scales come from a real problem and are defined, e.g., by the DM. However, in ORCLASS the set of alternatives is formed as all possible combinations of values on the scales of criteria. This may cause redundant calculations when the set of all possible alternatives is large and only some alternatives should be used as real cases.

Instead of real values, the ranks of values on the scales of criteria are used in the method for all calculations. Operating with ranks allows avoiding transformation of criteria values to numbers in the verbal methods. We will see below how this is realized. The method proposes inferring preferences directly from the DM in the form of answers to questions. For example: "Please, classify the alternative x_i to one of the possible classes in $L = \{l_1, \ldots, l_s\}$". However, it would require a considerable amount of time from a DM to classify every alternative from the set of possible alternatives. Moreover, cognitive loading of the DM (due to the limits of short-term memory) grows with the increasing number of possible alternatives, number of criteria, number of values on them, or number of classes will produce more questions.

Researchers have been looking for ways to overcome this problem. They discovered that the assignment of some alternatives allows classifying several others according to the relations of preference between the alternatives and with regards to the order of classes. The property of an alternative to be able to classify indirectly some other alternatives is called informativeness. The informativeness of an alternative is calculated as a sum of products (calculated for each possible class of the alternative) of the number of indirectly classified alternatives and the probability of the alternative to be assigned to each class. Assignment of the most informative alternative at each iteration allows classifying the maximal number of alternatives indirectly with the minimal number of requests to the DM. In such a way, ORCLASS reduces the number of alternatives to be classified by the DM directly. Let us formulate the problem mathematically.

If $G = \{g_1, \ldots, g_n\}$ is the set of criteria, then the scale of values for each criterion is $S(g_j) = \{g_{j1}, \ldots, g_{jt}\}$ and the scale of ranks for the same criterion is $R(g_j) = \{1, \ldots, t\}$, where t is the number of values on the scale of criterion g_j. The set of all possible alternatives is defined automatically as the Cartesian product of scales of criteria values $X = S(g_1) \times S(g_2) \times \cdots \times S(g_n)$. The number of classes $L = \{l_1, \ldots, l_s\}$ is known before the classification starts. However, their bounds are formed during an interactive procedure with the DM. We are solving a maximization problem, where higher values are preferred.

When the classification starts, for each alternative $x_i \in X$ $(i = 1, \ldots, m)$ a set of admissible classes L_i is defined (the classes where the alternative can be assigned). We also use the notation L_i^{old} for defining the set of admissible classes at the previous iteration. At the very beginning, the set of admissible classes contains all classes $L_i = L_i^{old} = L$ and its size is $|L_i| = s$. During the interactive preference elicitation procedure between the method and the DM, the classification of some alternatives is done indirectly. In such a way, the set L_i will be reducing until for each alternative x_i only one admissible class is left and then $|L_i| = 1$. This means that each alternative from the set of possible ones is classified into exactly one class.

It is assumed that the "worst" alternative with the least desirable values on all scales is assigned to the class with the smallest rank on the scale of classes, for instance, $x_1 \in l_1$, and the "best" alternative with the most desirable values on all scales is automatically classified to the class with the largest rank on the scale of classes, $x_m \in l_s$. The ORCLASS method is based on two relations:

1. The strict preference relation between two alternatives determines that the alternative x_i is better to the alternative x_r, if it is strictly preferred on at least one criterion $g_j(x_i) P g_j(x_r)$ and indifferent on the rest of the criteria $g_f(x_i) I g_f(x_r)$, $j \neq f$. But here instead of real values of alternatives on scales of criteria their ranks are compared. For example, $r_j(x_i)$ and $r_j(x_r)$ are ranks of alternatives x_i and x_r on the scale of the criterion g_j. Consequently, for the strict preference of the alternative x_i when compared to the alternative x_r the following relations $r_j(x_i) P r_j(x_r)$ and $r_j(x_i) I r_j(x_r)$ are estimated. Then,

the strict preference relation between the two alternatives is defined as follows:

$$\begin{aligned}&x_i \text{ P } x_r \ (x_i \text{ is strictly preferred to } x_r),\\ &\text{if } r_j(x_i) \geq r_j(x_r) \text{ and } r_f(x_i) > r_f(x_r) \text{ on at least one criterion } g_f,\\ &f, j = 1, \ldots, n; j \neq f; i, r = 1, \ldots, m; i \neq r.\end{aligned} \tag{8}$$

However, this relation is not enough for considering all the situations that may appear in the decision aiding process. That is why the next relation is also used.

2. The ordering relation between alternatives from different classes determines, for instance, that alternatives belonging to the second class are preferred to alternatives in the first one and so on. If the alternative x_i belongs to the class l_q and the alternative x_r to the class l_k, where q and k are class ranks, and we know that the alternative x_i is preferred to the alternative x_r, then the class l_q should have a higher rank than the class l_k: $q > k$ (as we consider a maximization problem a higher rank of the class is preferred to a lower one). The following binary relation on the set of alternatives X assigned to some classes from the set L is established [100]:

$$\begin{aligned}&x_i \text{ O } x_r \ (x_i \text{ belongs to the class with a higher rank}\\ &\text{than the rank of the class where } x_r \text{ is located}),\\ &\text{if } x_i \in l_q, x_r \in l_k \text{ and } q > k.\end{aligned} \tag{9}$$

For a non-contradictory classification, the following relation should be fulfilled [100]:

$$\begin{aligned}&\text{if } x_i \text{ P } x_r \text{ is true,}\\ &\text{then } x_r \text{ O } x_i \text{ cannot be true.}\end{aligned} \tag{10}$$

This means that alternatives from a class with a lower rank cannot be preferred to the alternatives from a class with a higher rank.

As mentioned earlier, the ORCLASS method is based on the idea of selecting the most informative alternative and proposing it to the DM for classification at each iteration. For calculation of the informativeness of an alternative, the following concepts are used: the center of a class (empty or non-empty), the distance between two alternatives, the size of a set of possible classes, the sum of distances between any alternative and the center of each class, the number of indirectly assigned alternatives, and the probability of an alternative to be assigned to a particular class. Let us consider the definitions of all these concepts.

The *center of a class* is defined in a different way for empty and non-empty classes. For a *non-empty class* l_q, the center x_q represents a fictitious alternative with average ranks of criteria values of the alternatives assigned into this class, $x_i \in l_q$. Then, each criterion value $g_j(x_i)$ of the center x_q of the non-empty class l_q is taken on each scale of criterion g_j ($j = 1, \ldots, n$) with regards to the rank $r_j(x_q)$ computed according to the expression:

$$r_j(x_q) = \frac{\sum_{x_i \in l_q} r_j(x_i)}{|l_q|},$$

where $|l_q|$ is the size of the set of alternatives that are assigned to the class l_q.

For an *empty class* l_q, the center x_q is calculated based on the assumption that there are non-empty classes l_w and l_u on both sides of the class l_q (not necessary next to this class but with indices $u < q < w$). Then, each criterion value $g_j(x_i)$ of the center x_q of the empty class l_q is taken on each scale of criterion g_j ($j = 1, \dots, n$) according to the rank calculated by the following formula:

$$r_j(x_q) = r_j(x_u) + \frac{r_j(x_w) - r_j(x_u)}{w - u}(q - u),$$

where $X_q = \emptyset$, $X_w \neq \emptyset$, $X_u \neq \emptyset$, and $u < q < w$ are the sets of alternatives of the classes l_q, l_w, l_u, respectively.

The *maximal distance between two alternatives* from the set X is defined as d_{max} and calculated according to the formula:

$$d_{max} = \max_{x_i, x_r \in X} \sum_{j=1}^{n} |r_j(x_i) - r_j(x_r)|,$$

where x_i, x_r are two alternatives from the set X.

The *distance between the alternative x_i and the center x_q of the class l_q* is defined as d_{iq} and is computed as follows:

$$d_{iq} = \sum_{j=1}^{n} |r_j(x_i) - r_j(x_q)|.$$

The *distance between the alternative x_i and the center x_q of each class l_q* from the set of admissible classes L_i for this alternative is $\sum_{l_q \in L_i} d_{iq}$.

The *number of indirectly assigned alternatives s_{iq}* is estimated for each alternative x_i when it is assumed that this alternative is assigned to the class l_q. Then, we have to calculate the strict preference relations with regards to (8) $x_i \mathrm{P} x_r$ and $x_r \mathrm{P} x_i$ between this alternative x_i and each not yet classified alternative x_r such that $r \neq i$. Based on the relation of order between classes according to (9), it is possible to reduce the set of admissible classes L_i for the not yet classified alternatives. For some of them, this set will be equal to one $|L_i| = 1$. That means that those alternatives are classified indirectly. Thus, we can calculate such number for each alternative x_i if we assume or if the DM states that it is assigned to the class l_q.

The *probability p_{iq} of the alternative x_i to be assigned to the class l_q* is calculated according to the following formula:

$$p_{iq} = \frac{d_{max} - d_{iq}}{|L_i| d_{max} - \sum_{l_q \in L_i} d_{iq}}.$$

Then, for each alternative x_i from the set X, the *informativeness* Φ_i is calculated as the sum of products (calculated for each class, to which this alternative can be assigned $l_q \in L_i$) of indirectly classified alternatives s_{iq} and the probability of each alternative to be assigned to the particular class p_{iq}:

$$\Phi_i = \sum_{l_q \in L_i} p_{iq} s_{iq}. \tag{11}$$

This informativeness index is calculated for each alternative to be assigned into each admissible class.

The algorithm of the ORCLASS method

Step 1 Define the subset $X_L \in X$ of alternatives that are not yet classified by checking the sizes of the sets of admissible classes L_i for each alternative x_i $(i = 1, \dots, m)$. If we have $|L_i| = 1$, then the alternative x_i is classified. If there are no alternatives with more than one admissible class in the set of not yet classified ones $X_L = \emptyset$, then the classification is finished. If there are alternatives, for which $|L_i| > 1$, the procedure of classification continues by moving to *Step 2*.

Step 2 Calculate informativeness for each alternative x_i from the set of not yet classified alternatives X_L according to the formula (11).

Step 3 Select the alternative x_i with the maximal informativeness such that:

$$\Phi_i = \max_{x_r \in X_L} \Phi_r.$$

Step 4 Present this alternative x_i to the DM for direct classification. Ask the DM to select the class l_q $(q = 1, \dots, s)$ for the presented alternative x_i.

Step 5 Check the existence of inconsistencies. Inconsistencies occur if the DM assigns the alternative x_i to a class l_q that does not belong to the set of admissible classes L_i. In this situation, all inconsistencies should be eliminated according to the procedure described below.

Step 6 For each not yet classified alternative $x_r \in X_L$, recalculate the subset of admissible classes L_r (that has been L_r^{old} at the previous iteration) based on the information obtained from the DM at the previous step (where in our case x_i has been assigned to l_q) according to the formula:

$$\begin{array}{l} \text{if } x_i \in l_q \text{ and } x_i \text{ P } x_r, \text{ then } L_r = L_r^{old} \cap l_1, \dots, l_{q-1}, \\ \text{if } x_i \in l_q \text{ and } x_r \text{ P } x_i, \text{ then } L_r = L_r^{old} \cap l_q, \dots, l_s, \\ L_i^{old} = L_i. \end{array} \tag{12}$$

After such modifications of the set of admissible classes L_r for each alternative $x_r \in X_L$, some alternatives in the set of not yet classified ones may happen to have only one class in a set of admissible classes (for instance, for x_r, $|L_r| = 1$). These alternatives are then classified indirectly. At the end of this step, the iteration is finished and a new one should be started in *Step 2*.

In case of two classes, there are no inconsistencies because once the DM has assigned an alternative to a class, all other alternatives with not less preferable values upon all criteria will be assigned to the same class. For the problems with more than two classes, the information inferred from the DM is checked for consistency with the previously obtained information according to the property (10). For example, if $q < k$ and the DM assigns the alternative x_i to the class l_q and at the next iteration the alternative x_r to the class l_k, and $x_i P x_r$, a contradiction between the previously made classification and the current one appears, since one of the two conditions presented below is violated:

$$\begin{array}{l} \text{if } x_i \in l_q \text{ and } x_r \in l_k \text{ and } x_i \text{ P } x_r, \text{ then for } q > k \text{ classification is correct,} \\ \text{if } x_i \in l_q \text{ and } x_r \in l_k \text{ and } x_r \text{ P } x_i, \text{ then for } q < k \text{ classification is correct.} \end{array} \tag{13}$$

The reasons for the appearance of inconsistencies are the following: the DM commits errors in the classification or the initial values on the criteria are estimated incorrectly. The second case is difficult to trace and it is easier to prevent it by checking completeness and other properties of the set of criteria (see Section 2.1.4). The first kind of error is common for any DM due to human fallibility. It may be connected to losing attention or absence of a straight policy of classification and acting by trial and error; it can also depend on the limited capacity of the human memory. In this case, the DM should think over his or her answers and resolve inconsistencies with a new classification of the same alternative.

When all the alternatives from the Cartesian product are classified, the ORCLASS method searches for the boundary alternatives between classes and using boundary alternatives constructs the classification rule or rules for each class. The boundary alternative can be detected when all the alternatives from the Cartesian product of all criteria values are classified with regards to the following property of boundary alternative. Indeed, for the boundary alternative, changing value on at least one criterion moves it to a neighboring class. The obtained set of rules is universal for the set of criteria and can be used for any set of given alternatives defined on the same set of criteria and scales for them.

Based on the obtained set of rules, it is possible for the DM to obtain an explanation for the assignment of each alternative. The explanation of classification is presented as a location of the alternative regarding to the boundary alternatives.

The effectiveness of the ORCLASS method has been checked in the literature [100] according to the statistical modeling of different classification problems by variation in the number of criteria, the number of values on scales of criteria, and the number of classes. The experimental results of statistical tests are presented in Table 1 adapted from [100]. Here, the average number of questions proposed to the DM in order to realize a complete classification with the ORCLASS method is obtained experimentally for different sets of alternatives with the following sizes: 81, 256, 243 or 1024, estimated on 4 or 5 criteria, and classified into 2, 3 or 4 classes.

TABLE 1 Average number of questions for classification by ORCLASS

Size of G	Number of values on the scales of criteria	Size of A	Size of L		
			2	3	4
4	3	81	**8**	**13**	**17**
	4	256	**10**	**14**	**21**
5	3	243	**10**	**18**	**25**
	4	1024	**14**	**24**	**33**

However, some limitations of ORCLASS may be observed. The results of statistical tests, provided in [95], show that the ORCLASS method works efficiently only on small sets of alternatives with no more than 5 criteria and with 4 values in each criterion classified into 4 classes. For larger problems, the recalculation of the informativeness of each alternative at each iteration of a dialog between the DM and preference elicitation procedure is complex and, consequently, time-consuming. The other feature of ORCLASS is the classification of the whole set of alternatives obtained as the Cartesian product of scales of criteria. However, in real life, the DM is typically interested only in a given set of alternatives. In cases like that, the method does redundant calculations. Another drawback is the usage of an absolute value of informativeness as a measure, while in practice, relative estimation is more natural (for details, see [95]). The last two shortcomings are eliminated in the SAC method.

In the SAC method, the set of alternatives to be classified is given by the DM (allowing a reduction in the number of calculations if the set of real alternatives is smaller than the set of alternatives obtained as the Cartesian product of scales of criteria). However, alternatives obtained as the Cartesian product of scales of criteria are still used in the SAC method whenever they have a higher informativeness (and allow classifying indirectly a larger number of alternatives) when compared to the given alternatives.

Another improvement of SAC when compared to ORCLASS lies in considering a variation of the number of indirectly classified alternatives (for more details, see [95]). Let us consider an example: the most informative alternative has the most desirable values on all criteria but one, on which it has the value that is different from the most desirable value on one rank (on the scale of criterion). In a situation, where this alternative is assigned to a class with the smallest rank, a great number of alternatives is classified indirectly. Even though the probability of such an assignment is small, and in real life the impossibility of such a situation can be seen with an unaided eye, in the method this probability still can be large enough to make our assumption valid and such an alternative to be the most informative one. Thus, this alternative will be provided to the DM for classification and he or she assigns it to the class with the highest rank that will not be of any help for the indirect classification. This kind of a situation is not taken into account in the ORCLASS method because it considers the absolute average (or mean) as a measure of informativeness. Instead, in the SAC method, a relative estimation (or variation) σ is taken [100]. Then, informativeness is calculated as:

$$\tilde{\Phi}_i = \frac{\Phi_i}{1 + \beta \frac{\sigma}{\Phi_i}} = \frac{\Phi_i}{1 + \beta \frac{\sqrt{\Sigma_{l_q \in L_i} p_{iq} s^r_{iq} - \Phi_i^2}}{\Phi_i}}.$$

Here, Φ_i is the informativeness as it is calculated in the ORCLASS method; σ is the variation of informativeness; s^r_{iq} is the number of indirectly classified alternatives obtained as a result of assigning the alternative x_i to the class l_q and calculated in the same way as in ORCLASS with such a difference that only the given (real) alternatives $X^r = \{x^r_1, \ldots, x^r_m\}$ are taken into account; and β ($\beta > 0$) is the relative importance of variance between the number of indirectly classified alternatives for different classes. It shows the influence of the variation on the informativeness. When variation is not taken into account $\beta = 0$, then informativeness is calculated in the same way as in ORCLASS but for the given alternatives.

Thus, the difference between ORCLASS and SAC is in the calculation of the informativeness for each alternative at each iteration and in the taking into account only given alternatives. The rest of the alternatives can be used because at some iterations they may happen to be the most informative ones and, consequently, allow assigning the maximal number of given alternatives indirectly. The relative character of informativeness is also taken into account. Recommendations for the values of variation to be used in different problems have been studied in [81] and are collected in Table 2. The rest of the classification procedure is the same as in ORCLASS.

TABLE 2 Recommendations for values of variance in SAC

Size of G	Size of L		
	2	3	4
4	**2.4**	**2.3**	**2.2**
5	**3**	**2.7**	**2.4**
6	**3.5**	**3.1**	**2.6**

Statistical tests provided in [95] show the advantage of SAC when compared to ORCLASS. The efficiency of SAC and ORCLASS is estimated as a number of questions Q to the DM when classifying the set of given alternatives X^r. It is also possible to compare the number of questions posed to the DM when a direct classification of each alternative is organized. In this case, the number of questions is naturally equal to the set of alternatives. One can say that this is the worst case.

As mentioned before, in ORCLASS and in the same way in SAC, the alternatives that lie on the boundaries of the classes are defined in the framework of the methods. When such alternatives are known, the assignment of any alternative from the set of given alternatives to the class can be done by comparison to boundaries of each class. Theoretically, the minimal number of questions posed to a DM in the VDA method would be questions about alternatives that lie on the boundaries, but that is an ideal case and it does not work in practice. However, the efficiency of the VDA method can be estimated with the help of the ideal case.

One more VDA method called CYCLE [93] has appeared recently. The method works for initial and output conditions that are similar to those of the ORCLASS method. However, it improves the step of selection of alternative to be posed to the DM. Indeed, it builds "chains" of alternatives between two alternatives that are assigned to two different classes. The chains are constructed in such a way that two subsequent alternatives in the chain differ on one criterion value. Then, for each chain a subset of alternatives equidistant from the two limit alternatives of the chain is selected. For the alternatives from this subset, the simplified version of informativeness is calculated (for details, see [93]). Then, the alternative with maximal informativeness is posed to the DM for direct classification.

The CYCLE method is reported to outperform ORCLASS and be efficient when compared to the monotone function decoding algorithm on smaller data sets and less effective on more complicated problems (for details, see [93]). Even though this direction seems to be promising, many open questions remain. For instance, how to select two alternatives for the construction of chains. It is clear that at the very beginning we have two alternatives with all the worst and all the best criteria values classified to the worst and best classes, respectively. However, there is no description about the selection of the two alternatives for the chain construction on the next iterations. Here, another question may appear, i.e., whether it is preferable to construct chains between alternatives from the closest classes or from the remotest ones? On the other hand, it may be possible to construct several chains between two alternatives from different classes, and there is no description about which one to select in this case. The method works for the Cartesian product and still may be adapted to the set of given alternatives (in a similar way it is done in SAC).

3.5.2 Concluding Remarks

Both VDA methods for classification, ORCLASS and SAC, require similar information to be defined by the DM: the criteria with scales of the values (that can be verbal or numerical) and the number of classes as well as their order. In ORCLASS, the set of given alternatives is composed as a Cartesian product of all criteria scales, while in SAC the set of given alternatives should be specified. The methods are parameter-free and do not require any parametric information and/or assignment examples to be defined by the DM. On the contrary, the methods select some (but not all) alternatives from the set of alternatives composed as the Cartesian product of all criteria scales. In such a way, the methods try to reduce the cognitive loading of the DM. However, the alternatives posed to the DM for direct classification are usually very difficult cases and located in the near-boundary regions. That is why the DM may be loaded cognitively even with a small number of alternatives that he or she should classify directly and may be inconsistent when classifying such alternatives. The methods check the consistency of each classification determined by the DM with the previous ones and ask the DM to rethink his or her answers in case of contradictory classifications. Resolving of a contradiction may also load the DM cognitively; however, such exercising of direct classification may help the DM to understand his or her preferences. The methods search for the boundary alternatives between classes and, based on them, may provide explanation about the appearance of each alternative in the class.

The advantage of the methods is an absence of need for the transformation of verbal values into numerical analogies at the input or output of the methods. Moreover, the VDA methods keep the transparency of the verbal values during the iterative process of DM's interrogation and pose alternatives to be classified with verbal values on the criteria. However, there is no way to treat attributes in VDA methods.

As a weak feature of the existing VDA methods, we consider the heuristics embedded in the calculation of the informativeness index. Indeed, when calculating the probability of each alternative to be assigned in a class, it is assumed that the center of an unknown class is located with an equal distance from the centers of already known classes. This means that the centers of classes are uniformly distributed. However, in real-life this may not be the case. For instance, for the real-world examples described in [114], the distributions of alternatives between the classes are not always uniform. When developing the new Dichotomic Classification method presented in the Section 4.2 we try to take this issue into account.

The SAC method decreases the number of calculations when compared to the ORCLASS method [95]: only a given set of alternatives is classified, also relative informativeness is taken into account, but not the absolute one (for more details, see [95]). Even though the alternatives from the whole Cartesian product participate in the classification, and some of the alternatives that are not in the set of given alternatives (but at some iteration appear to be the most informative) may be asked from the DM. In addition, there is still the evident dilemma. The effectiveness of the VDA methods considerably decreases when increasing the number of criteria or the values on the scales of criteria, while with increasing the number of classes the effectiveness increases [95]. The observations on the limitations of human information processing system [92] recommend a problem with a maximum of 10 classes and a maximum of 15 criteria, while the number of values on the scales of criteria should not exceed 10 when applying the SAC method. However, in the real world, still people must solve problems of larger size. The new method proposed in Section 4.2 improves the efficiency of verbal classification and aims at being able to cope with bigger size problems.

Because VDA methods assume that the DM's knowledge and preferences may not be very stable and may change over time, they perform consistency tests in order to check contradictory classifications made by the DM during the interactive interrogation. On the one hand, giving to the DM information about his or her own contradictory classifications may help him or her to learn about once's preferences. On the other hand, VDA methods force the DM to resolve contradictory classifications without letting him or her be imprecise, and, what is more crucial, without supporting and guiding him or her towards resolving inconsistencies. However, there are possibilities for improving this part of VDA by studying human behavior. (For instance, in a similar way it it done for the choice type of MCDA problems in [51].)

3.6 Fuzzy Sets

As has been mentioned above, uncertainty is treated differently in various MCDA methods. Fuzzy sets developed by Zadeh [166] and adapted to MCDA in [15] look into uncertainties when defining human preferences, in particular, in verbal terms. The concept of fuzzy sets is used for the definition of imprecise belonging

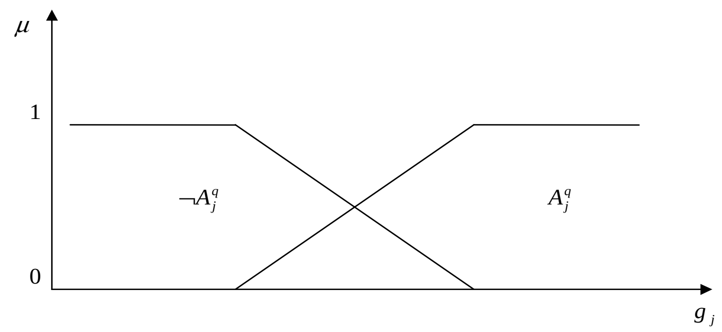

FIGURE 10 Partial membership function for fuzzy set

of an alternative to some set. For a classification problem, it is assumed that there is some partial knowledge about belonging of an alternative to one or several classes, however, such knowledge is not crisp and is represented by fuzzy sets. Then, whether the alternative x_i belongs to the class l_q should not necessary be true or false but may be partially true. The degree of truth is called *membership degree*. It is defined on the interval $[0, 1]$ with 0 indicating that a proposition is false and 1 that it is true, and that there is a partial degree of truth in between.

Another specific feature of fuzzy sets is the possibility to take into account not only criteria but attributes (see Section 2.1.4). Let us remember that when compared to the criteria the attributes are neither minimized nor maximized and the scales for them are not ordered. For the classification problem, a partial membership function shows the degree to which an alternative estimated on a criterion or attribute belongs to a class.

The usual way to present a fuzzy membership function μ for the proposition A_j^q: "x_i belongs to the class l_q according to the criterion or attribute g_j" is shown in Figure 10. Here, the complementary part for the negation of the proposition $\neg A_j^q$:"x_i does not belong to the class l_q according to the criterion or attribute g_j" is also shown. The partial fuzzy membership functions are defined by the DM. It is assumed that he or she is able to estimate criteria and attributes values and define such of them that support and/or oppose assignment of alternatives to each class. This task may be difficult and cognitively loading for the DM.

The overall membership function is constructed as an aggregation of the partial membership degrees on each criterion or attribute. The aggregation is constructed with fuzzy rules that may be defined as follows:

$$\text{if } A_1^q \wedge \cdots \wedge A_n^q, \text{ then } x_i \in l_q,$$

where the fuzzy value of membership function A_j^q corresponds to each value of criterion or attribute g_j when assigning the alternative x_i to the class l_q.

The main ideas of fuzzy sets are very simple, but the details can be very specific as well as complicated membership functions can be used. We are not going into more detail that can be found, for instance, in [27].

Modeling the uncertainty connected to human preferences with fuzzy sets is promising and appealing for the developers of MCDA because of the simple ideas behind it. Moreover, in [37], it is mentioned that in many works, including [8], [68], [69], [70], [126], the fuzzy rules have been developed for classification problems. In particular, fuzzy sets have been used for modeling ordinal preferences expressed in verbal terms. In addition, fuzzy sets are applied for modeling of imprecision in crisp theories, such as outranking methods. Another direction of fuzzy sets development is their hybridization with other well-known techniques such as neural networks (neuro-fuzzy system), expert systems (fuzzy rule-based expert systems), and mathematical programming (fuzzy mathematical programming) [37].

Despite of such a large application area of fuzzy sets, some practical problems in the implementation have been noticed [16]. First, the choice of the shape of the fuzzy membership function is difficult. For instance, in order to define such shape the DM has to be able to estimate the criterion or attribute values that support assignment of the alternatives from the set of given alternatives into each class. On the other hand, when membership degree is estimated, the explanation of reasons for it may be difficult to interpret for the DM. Thus, the classification results in the following form: "the alternative x_i has greater membership than the alternative x_r to be assigned to the class l_q", may be not evident for the DM. That is why most often a linear function between two extremes on the interval of criteria or attributes values is selected as a membership function. Secondly, the operations of *union* and *intersection* applied in fuzzy rules for operating with membership values have different meanings [167]. Thus, more complicated operations have to be applied, but at the moment simple algebraic expressions are used "by default".

3.7 Rough Sets

Originally, the rough set approach (RSA) was proposed by Pawlak [129] as a tool for the analysis of sets which boundaries are not precise. At the beginning, the rough set theory was developed for nominal classification [130] and later adapted to solve ordinal classification [55], [59]. The idea is to present a class as a rough set that contains alternatives that definitely belong to the class (that is, so-called lower approximation of the class) and alternatives that could belong to the class (that is, so-called upper approximation of the class). When compared to the fuzzy sets approach, where the knowledge about whether an alternative belongs to a particular set or not is a source of uncertainty estimated by a membership function degree, in rough sets, the class (that is, collection of alternatives) is considered to be uncertain.

In the original RSA developed for nominal classification, the given set of assignment examples $X^* \in X$ (where X is the set of alternatives) is estimated with the set of attributes G and have to be assigned to the set of classes L. The attributes may have numerical or verbal values. In RSA adapted for ordinal classification, the possibility of estimation on the criteria is added and the order of classes is known. The classification is based on the set of assignment examples provided by the DM. Such decisions he or she is used making in the past; or decisions made for realistic alternatives not considered in the set of given alternatives; or decisions for a limited set of alternatives from the set of given alternatives. The idea is to derive a set of rules in the form of the proposition: "*if* ..., *then* ..." from assignment examples.

The development of RSA starts from the assumption that with every assignment example provided by the DM some knowledge, so-called granules, are connected. These are blocks of alternatives indifferent with respect to a subset of criteria and/or attributes. In rough set terminology, criteria and attributes are considered to be condition attributes, and classes to which such alternative belongs are decision attributes.

RSA also works for the ranking and choice types of MCDA problems [57]. The brief overview presented in this section is based on the comprehensive surveys [56] and [60]. Next, we consider RSA for nominal classification problem and its modification for ordinal classification problem.

3.7.1 Rough Sets Approach for Nominal Classification

In nominal classification, for the definition of similarity between any two alternatives a so-called indiscernibility relation that is the indifference relation in MCDA (see Section 2.1.5) is applied. Such relation is estimated for all pairs of assignment examples provided by the DM. In RSA, indifference may be defined on some subset $G_I \subseteq G$ of attributes sufficient for considering these alternatives to be indifferent. Later, the similarity relation was proposed in [146] that can also be used in rough sets for nominal classification [61].

The algorithm of Rough Sets Approach for nominal classification

Step 1 Build the granules of knowledge about a given set of assignment examples with lower (precise) and/or upper (approximate) approximations of the set of unordered classes. This is done by means of defining blocks of alternatives indifferent with respect to a subset of attributes. The lower approximation $\underline{G_I}(X_{l_q})$ defines all alternatives X_{l_q} from the given set of assignment examples X^* $(i = 1, \ldots, m^*)$ which definitely belong to the class l_q $(q = 1, \ldots, s)$ with respect to the subset $G_I \subseteq G$ of attributes, while upper approximation $\overline{G_I}(X_{l_q})$ contains all alternatives X_{l_q} that possibly belong to the class l_q with respect to the subset $G_I \subseteq G$ of attributes. For estimation of lower and upper approximations, it is necessary to calculate the indifference relation $I_{G_I}(x_i, x_r)$ between each pair of alternatives x_i and x_r from X^* only for a subset of attributes $G_I \subseteq G$ that is considered to be sufficient enough for the indifference between two alternatives to be valid:

$$\underline{G_I}(X_{l_q}) = \{l_q \in L, x_i, x_r \in X^* : I_{G_I}(x_i, x_r) \subseteq X_{l_q}\},$$
$$\overline{G_I}(X_{l_q}) = \{\bigcup_{l_q \in L, x_i, x_r \in X^*} I_{G_I}(x_i, x_r)\}.$$

Here, the indifference relation $I_{G_I}(x_i, x_r)$ is calculated as follows:

$$\begin{aligned} &I_{G_I}(x_i, x_r) = \{x_i, x_r \in X^* : \\ &g_j(x_i) = g_j(x_r) \text{ for any attribute } g_j \in G_I, G_I \subseteq G\}, \end{aligned} \tag{14}$$

where g_j $(j = 1, \ldots, n)$.

Step 2 Define the boundary "doubtful" region for each class l_q as a difference between the upper and lower approximations:

$$BG_I(X_{l_q}) = \overline{G_I}(X_{l_q}) - \underline{G_I}(X_{l_q}). \tag{15}$$

According to the upper approximation, the alternative may be assigned to several classes, and, thus, the classification is ambiguous.

Step 3 Estimate the quality of classification of the set of given alternatives X^* into the classes L using each subset of attributes $G_I \subseteq G$:

$$\gamma_{G_I}(X^*) = \frac{\sum_{q=1}^{s} |\underline{G_I}(X_{l_q})|}{|X|}.$$

It shows the ratio of the correctly classified assignment examples with respect to the subset of attributes $G_I \subseteq G$ when compared to the set of all assignment examples. The idea is to define the quality of classification based on each subset of attributes $G_I \subseteq G$ and find a minimal subset that guarantees a high quality of classification. In an ideal case, the quality of classification with such subset should be the same as with the whole set of attributes, i.e., $\gamma_{G_I}(X^*) = \gamma_G(X^*)$. Such a subset is called a *reduct*. There can be several reducts; their intersection is called a *core*.

It is also possible to estimate the degree with which each particular alternative belongs to each class using a so-called *membership degree*:

$$\mu_{X_{l_q}}^{G_I}(x_i) = \frac{|X_{l_q} \subseteq I_{G_I}(x_i, x_r)|}{|I_{G_I}(x_i, x_r)|}.$$

Such a membership function is close to conditional probability and shows the degree of confidence calculated from the available data (when compared to the fuzzy sets where membership function is assumed to be of a particular form defined by the DM).

Step 4 Develop the set of rules for the assignment examples and selected subset of attributes with regards to the selected strategy. The "*if* ...," part of the rule is called *conditional* and "*then* ..." part is called *decisional*. The rules are of the following form:

$$\text{if } (g_1(x_i) \text{ is } V_{g_1}) \wedge (g_2(x_i) \text{ is } V_{g_2}) \wedge \cdots \wedge (g_n(x_i) \text{ is } V_{g_n}), \text{ then } x_i \in l_q,$$

where $g_j(x_i)$ is value of the attribute g_j that take values V_{g_j} on the scale of attribute values Sg_j.

There are several strategies according to which the set of rules for the given set of assignment examples may be inducted in different RSA methods (for more details, see [56], [60]). The most often used strategy assumes that a *minimal* set of rules that covers all assignment examples is discovered. For a rule from the minimal set, removing at least one attribute from the conditional part of the rule will result in extension of this rule to a subset of alternatives that are not assignment examples.

Step 5 Check inconsistencies in the assignment examples that may occur when indiscernible assignment examples are assigned into different classes. Such situations should be taken into account and may be resolved using RSA. We say that the rule *covers* the assignment example if the description of the alternative matches the conditional part of the rule, and that the rule *supports* the assignment example if the description of the alternative matches both the conditional and decisional parts of the rule. The following situations may appear:

1. The assignment example is covered by only one rule.
2. The assignment example is covered by several rules and is assigned into the same class.
3. The assignment example is covered by several rules and is assigned into different classes.
4. The assignment example is not covered by any rule.

The classification in the first two situations is obvious. The third conflicting situation is resolved if the strength of each rule is taken into account [63]. The *strength* of the rule defines the number of assignment examples that support the rule. There have been different ways to resolve the fourth situation. Most of them apply the following approach: searching for the rules that cover the assignment example partially [63]. Both of the approaches also consider the strength of rules.

3.7.2 Rough Sets Approach for Ordinal Classification

RSA has been adapted for ordinal classification in [55]. Recently, RSA for ordinal classification has been extended into Dominance-Based Rough Set Approach (DRSA) [58].

The ordinal character of a problem requires the introduction of an additional relation: outranking relation (see Section 2.1.5). In RSA, the outranking relation may be defined on some subset $G_I \subseteq G$ of criteria sufficient for such relation to be valid:

$$\begin{array}{l} x_i \; S_{G_I} \; x_r \; (x_i \text{ outranks } x_r), \\ \text{if } g_j(x_i) \; S \; g_j(x_r) \text{ for all criteria } G_I \subseteq G, \\ i, r = 1, \ldots, m^*; i \neq r. \end{array} \tag{16}$$

The verbal values can be considered in RSA similarly to VDA. The attributes are compared based on the indifference relation (14). Next, we describe how, with the help of these relations, the classification is done.

The algorithm of Dominance-Based Rough Set Approach for ordinal classification

Step 1 Construct the outranking relation (16) for all pairs of assignment examples $x_i, x_r \in X^*$ $(i = 1, \ldots, m^*)$. Then, the following binary relation on the set of assignment examples from X^* is estimated (similarly to relation (9) but for the subset of attributes and/or criteria $G_I \in G$):

$$\begin{aligned}&x_i\ O_{G_I}\ x_r\ (x_i \text{ belongs to the class with a higher rank}\\&\text{than the rank of the class where } x_r \text{ is located}),\\&\ \text{if } x_i \in l_q, x_r \in l_k \text{ and } q > k.\end{aligned}$$

Step 2 Using information about order of classes, for each class l_q, it is possible to define the upper $L_q^{\geq}$ and lower $L_q^{\leq}$ unions of upward and downward ranked classes:

$$\begin{aligned}L_q^{\geq} &= \textstyle\bigcup_{t \geq q} l_t,\\ L_q^{\leq} &= \textstyle\bigcup_{t \leq q} l_t.\end{aligned}$$

For each assignment example $x_i \in X^*$, the sets $X_{G_I}^{+}(x_i)$ and $X_{G_I}^{-}(x_i)$ of alternatives to which the alternative x_i is preferred and which are preferred to the alternative x_i, respectively, are defined with respect to the selected set of attributes and/or criteria $G_I \in G$:

$$\begin{aligned}X_{G_I}^{+}(x_i) &= \{x_r \in X^*, i \neq r : x_i S_{G_I} x_r\},\\ X_{G_I}^{-}(x_i) &= \{x_r \in X^*, i \neq r : x_r S_{G_I} x_i\}.\end{aligned}$$

In the ordinal classification DRSA method, the set of possible classes should be defined for each assignment example. It is done in the same way as in VDA (12); however, with respect to the selected subset of attributes and/or criteria $G_I \in G$:

$$\begin{aligned}&\text{if } x_r \in l_k \text{ and } x_i\ S_{G_I} x_r, \text{then } X_{G_I}^{-} \cap L_q^{\geq} \neq 0,\\&\text{if } x_r \in l_k \text{ and } x_r\ S_{G_I} x_i, \text{then } X_{G_I}^{+} \cap L_q^{\leq} \neq 0.\end{aligned} \tag{17}$$

Step 3 For the ordinal case, the upper and lower approximations are built for upward $L_q^{\geq}$ and downward $L_q^{\leq}$ unions of possible classes on the sets $X_{G_I}^{+}(x_i)$ and $X_{G_I}^{-}(x_i)$ of alternatives to which the assignment example x_i is preferred and which are preferred to the assignment example x_i, respectively, in the following way:

$$\begin{aligned}\underline{G_I}(L_q^{\geq}) &= \{x_i \in X^* : X_{G_I}^{+} \subseteq\ L_q^{\geq}\},\\ \overline{G_I}(L_q^{\geq}) &= \textstyle\bigcup_{x_i \in L_q^{\geq}} X_{G_I}^{+} = \{x_i \in X^* : X_{G_I}^{-} \cap L_q^{\geq} \neq 0\}.\end{aligned}$$

$$\begin{aligned}\underline{G_I}(L_q^{\leq}) &= \{x_i \in X^* : X_{G_I}^{-} \subseteq\ L_q^{\leq}\},\\ \overline{G_I}(L_q^{\leq}) &= \textstyle\bigcup_{x_i \in L_q^{\leq}} X_{G_I}^{-} = \{x_i \in X^* : X_{G_I}^{+} \cap L_q^{\leq} \neq 0\}.\end{aligned}$$

Step 4 The boundary regions for the upper and lower approximations of classes are calculated in the same way as in the original RSA for nominal classification according to formula (15).

Step 5 Then, for each class the quality of classification of the set of assignment examples X^* into the class l_q using the subset of attributes and/or criteria

$G_I \subseteq G$ is computed:

$$\gamma_{G_I}(X_{I_q}) = \frac{|X^* - \bigcup_{2 \geq q \geq s} BG_I(L_q^{\geq})|}{|X|} = \frac{|X^* - \bigcup_{1 \geq q \geq s-1} BG_I(L_q^{\leq})|}{|X^*|}.$$

Reducts, core, and membership degree can be calculated in a similar way as in the original RSA method for nominal classification but only for the set of possible classes defined with expression (17).

Step 6 The decision rules are derived from the upper and lower approximations of possible classes in a similar way it is done for the original RSA method for nominal classification.

Step 7 For ordinal classification with DRSA, the possible inconsistencies are checked based on the outranking relation for the assignment examples from ordered classes in the same way as in VDA (13) but with respect to the subset of attributes and/or criteria $G_I \in G$:

$$\begin{aligned} &x_i \in L_q^{\geq}, \text{ but } X_{G_I}^{+} \cap L_{q-1}^{\leq} \neq 0, \\ &x_i \ni L_q^{\geq}, \text{ but } X_{G_I}^{-} \cap L_q^{\geq} \neq 0. \end{aligned}$$

Information about such inconsistencies should still be taken into account because, as stated in [137], such information may indicate limited discrimination power of attributes or show hesitation of the DM.

3.7.3 Extensions and Applications

Recently, DRSA [58] has been extended into variable-consistency dominance-based rough set approach (VC-DRSA) [61] and its modification [17]. The modification of DRSA method [17], in addition to taking into account the positive assignment examples (that are good representatives of classes), considers the negative ones as well (defining such alternatives that cannot belong to the class). Such an approach is reasonable, especially, when ambiguous assignments are considered.

An extensive list of applications of rough sets for classification was presented in the EURO XXII conference in Prague [62] in 2007, including financial portfolio decision analysis, credit rating and credit assessment problems, supporting facility service procurement, budget allocation in highway maintenance, and assessing rural sustainable development potentialities.

3.7.4 Concluding Remarks

Like all the methods based on the assignment examples, RSA suffers from a possible imprecision and incompleteness of the information provided by the DM. However, in RSA, there is a possibility for checking the inconsistencies that may appear in the information provided. In addition, there are possibilities for the analysis of importance and dependencies among criteria and hierarchical construction of criteria. The decisions taken under uncertainty and with partially missing values can be considered, and there is a possibility to model imprecise information by fuzzy sets [60].

In a similar way to VDA methods (see Section 3.5.1), rough sets do not require transformation of verbal values to numerical analogies at the input or output of the methods, and keep the transparency of the verbal values for the DM. In RSA for ordinal classification, the alternatives can be estimated on both the attributes and criteria. In addition, RSA methods for both, ordinal and nominal, classification types consider the possibility of classification based on subsets of criteria and/or attributes. This feature is very natural for human evaluations that is confirmed by experiments conducted in [96]. These show that usually experts and/or DMs develop a small number of rules with some combination of the most important criteria for the classification of alternatives.

3.8 Multicriteria Filtering Method

The Multicriteria Filtering (MC Filtering) method has been developed by Perny [131] based on the concordance and non-discordance principles that have also been used in the ELECTRE family of methods and presented in Section 3.1.1. The method allows ordinal and nominal classification of the set of alternatives X, estimated on the set of criteria G, into the set of classes L predefined with the set of boundary or reference alternatives B, respectively, for the ordered and not ordered classes. There can be several boundary or reference alternatives for each class l_q. In the case of ordinal classification, the MC Filtering method develop a so-called filtering procedure based on the so-called valued preference relation estimated for each pair of alternative x_i to be classified and the boundary alternative b_q^h. For nominal classification, the MC Filtering method constructs a filtering procedure based on the valued indifference relation.

The MC Filtering method for ordinal classification is very similar but more general than that of ELECTRE TRI presented in Section 3.1.1. The main difference is in the use of a valued preference relation (for details, see [131]) instead of an outranking relation for ordinal classification and a valued indifference relation (defined below) for nominal classification. In addition, it is possible to define several upper and lower boundary alternatives for each class. In this work, we present the MC Filtering method for nominal classification that was one of the first MCDA methods that consider classes to be not ordered.

The MC Filtering method assumes that the set of reference alternatives is available $B = \{b_1^1, \ldots, b_1^{t_1}, \ldots, b_s^1, \ldots, b_s^{t_s}\}$, where t_1 and t_s are the numbers of reference alternatives for the classes l_1 and l_s, respectively. For the estimation of the concordance and discordance indices, the outranking relations $x_i S b_q^h$ between each alternative x_i to be classified and each reference alternative b_q^h is calculated taking into account the indifference $q_j(\cdot)$, preference $p_j(\cdot)$, and veto $v_j(\cdot)$ thresholds as well as the weight w_j for each criterion g_j.

In MC Filtering, the exact meaning of weights is not specified, but, since the outranking relation is used throughout, we can suppose that it is the same as in the outranking methods. Then, the weight of the criterion indicates the relative importance of the criterion when compared to other criteria in terms of votes that support or oppose the assertion "one alternative is at least as good as another alternative". However, from the examples provided in [131], we can see that for ordinal classification the weight was taken with the same value for all criteria and classes, but for nominal classification there were different weights for each class. From this, we can guess that the authors assume that the weight for nominal classification may have a different meaning when compared to the ordinal one. Thus, the interpretation of weights here is possible: the weight w_j^q of the criterion g_j may indicate the relative importance of the criterion when compared to the other criteria that support the assignment in the class l_q. As in the ELECTRE TRI method, these may be specified in terms of votes and should be defined by the DM. Next, we present the algorithm of the MC Filtering method for nominal classification.

The algorithm of the MC Filtering method for nominal classification

Step 1 The partial outranking index $S_j(x_i, b_q^h)$ is calculated for each pair of the alternative to be classified x_i $(i = 1, \dots, m)$ and each reference alternative b_q^h $(h = 1, \dots, t_q)$ of each class l_q $(q = 1, \dots, s)$ on each criterion g_j $(j = 1, \dots, n)$ using the indifference $q_j(b_q^h)$ and preference $p_j(b_q^h)$ thresholds:

$$S_j(x_i, b_q^h) = \frac{p_j(b_q^h) - \min\{g_j(b_q^h) - g_j(x_i), p_j(b_q^h)\}}{p_j(b_q^h) - \min\{g_j(b_q^h) - g_j(x_i), q_j(b_q^h)\}}.$$

Step 2 The partial concordance index $C_j(x_i, b_q^h)$ is calculated for each pair of compared alternatives, each alternative to be classified x_i and each reference alternative b_q^h, as follows:

$$C_j(x_i, b_q^h) = C_j(b_q^h, x_i) = \min\{S_j(x_i, b_q^h), S_j(b_q^h, x_i)\}.$$

Step 3 The global concordance index $C(x_i, b_q^h)$ is obtained by aggregating the partial indices and taking into account the weights of the criteria:

$$C(x_i, b_q^h) = \sum_{j=1}^{n} w_j C_j(x_i, b_q^h),$$

where w_j is the weight of the criterion g_j. The weights are normalized and non-negative (see formula (3)).

Step 4 The partial discordance indices $D_j^S(x_i, b_q^h)$ for the outranking relation between two alternatives are calculated using the indifference $q_j(b_q^h)$, preferences $p_j(b_q^h)$ and veto $v_j(b_q^h)$ thresholds as follows:

$$D_j^S(x_i, b_q^h) = \min\{1, \max\{0, \frac{g_j(b_q^h) - g_j(x_i) - p_j(b_q^h)}{v_j(b_q^h) - p_j(b_q^h)}\}\}.$$

Step 5 The partial discordance index $D_j^I(x_i, b_q^h)$ for the indifference of two alternatives is calculated:

$$D_j^I(x_i, b_q^h) = D_j^I(b_q^h, x_i) = \max\{D_j^S(x_i, b_q^h), D_j^S(b_q^h, x_i)\}.$$

Step 6 The global discordance index $D(x_i, b_q^h)$ is considered as follows:

$$D(x_i, b_q^h) = 1 - \prod_{j=1}^{n}(1 - D_j^I(x_i, b_q^h))^{\alpha/n},$$

where α is a technical parameter that is used for modifying the degree of synergy between the criteria.

Step 7 Based on the concordance and discordance principles, the indifference relation $I(x_i, b_q^h)$ has been developed. It is valid if and only if the majority of criteria is in favor of this relation and there is no strong enough opposition to this relation in the minority of the criteria:

$$I(x_i, b_q^h) = \min\{C(x_i, b_q^h), 1 - D(x_i, b_q^h)\}.$$

Step 8 The membership index $d(x_i, l_q)$ for the alternative x_i to be classified into the class l_q is calculated based on evaluating the degree of indifference indices between this alternative and each reference alternative of the class.

$$d(x_i, l_q) = \max\{I(x_i, b_1^q), \ldots, I(x_i, b_{t_q}^q)\}.$$

Step 9 The alternative is assigned to the class if such a membership index has a maximal value:

$$x_i \in l_q \Longrightarrow d(x_i, l_q) = \max\{d(x_i, l_1), \ldots, d(x_i, l_s)\}.$$

The MC Filtering method is based on the concordance and non-discordance principles and stems from the outranking approach. Consequently, it inherits the advantages and disadvantages of this approach. Indeed, in the outranking approach, there is the possibility to consider two alternatives as not only indifferent or preferred but also to define them as weakly preferred or incomparable. The last two situations may appear if there are very conflicting criteria, and their differences on some criteria values cannot be compensated, or if the information provided by the DM is incomplete. Because some incompleteness may be involved in the model, the relations are not necessary transitive.

The definition of preference, indifference and, especially, veto thresholds in the MC Filtering method may be difficult. Moreover, as has been mentioned in [101], the application of such thresholds developed for modeling ordinal preferences is questionable in nominal classification, where there is no need to define preference relation but indifference or similarity only. Absence of the order between classes may also reduce the need for finding which of two alternatives is more preferable. That is why a calculation of the outranking index for the definition of indifference between two alternatives may look redundant. Another important issue is that of how to treat attributes. In [131], there is no clear differentiation between estimation on the set of criteria or attributes. However, the attributes, even if have an ordinal scales (but usually nominal ones), are neither to be maximized nor minimized; and the preference relation does not work for them. This issue should be clarified in MC Filtering.

3.9 Other Nominal Classification Methods

As has been mentioned in the introduction to this chapter, the first methods for nominal classification problems use classical approaches from statistics and artificial intelligence. Indeed, they assign the set of given alternatives into non-ordered classes based on the estimation of a similarity measure to the alternatives from a set of assignment examples. Usually, the alternatives are estimated on a set of attributes, which are neither minimized nor maximized. On the other hand, in MCDA theory, we have alternatives estimated on a set of conflicting criteria with ordinal scales of criteria values. That is why criteria and not attributes are considered in most of the MCDA methods (and in all methods presented in this section) for nominal classification. In an ideal case, for nominal classification methods, it would be good to be able to work with both criteria and attributes in the way it is done in rough sets (see Section 3.7.2).

In addition to criteria, in the nominal classification methods that stemmed from MCDA, the outranking or other preference relations are often used. It is typical for the modeling of ordinal preferences and is inherited from the MCDA methods for ordinal classification. However, the preference relation is usually estimated if the choice or ranking of alternatives should be done. On the other hand, the application of the outranking and preference relations for nominal classification problems may be arguable. In nominal classification, there is no need to compare which of two alternatives is preferred but to find how similar they are.

In next sections, we describe two methods developed recently for nominal classification in the framework of MCDA theory: PROAFTN [10] and TRINOMFC [101]. The PROAFTN method [10] calculates an indifference index based on the outranking relation for pairs of an alternative to be classified and a reference alternative in a way similar to that of MC Filtering. However, the reference alternatives for each class are considered to be fuzzy. Their values on each criterion are defined with the interval of possible values. The TRINOMFC method [101], on the other hand, substitutes an outranking relation by a similarity relation. The similarity function between two alternatives can be of different type for each criterion and should be defined by the DM. Next, we describe these two methods and discuss their advantages and disadvantages.

3.9.1 PROAFTN

The PROAFTN method was developed by Belacel [9], [10]. It is also based on the concordance and non-discordance principles of the outranking methods presented in Section 3.1. The method calculates a fuzzy degree of membership for each alternative x_i to be classified into each class l_q. Each alternative from the set of those to be classified X estimated on the set of criteria G is compared to each reference alternative from the set B that contains several reference alternatives for each class. This method assumes that each reference alternative $b_q^h \in B$ is presented in a fuzzy way: the values on each criterion g_j of this alternative are defined by an interval $[g_j^{min}(b_q^h), g_j^{max}(b_q^h)]$, such that $g_j^{min}(b_q^h) < g_j^{max}(b_q^h)$. Then, the relation of indifference between the alternatives defined in such a fuzzy way is called fuzzy indifference relation.

Because the intervals of possible criteria values of boundary alternatives are fuzzy, two positive discrimination thresholds, d_j^+ and d_j^-, should be defined by the DM. They indicate small differences, $g_j^{max}(b_q^h)$ and $g_j^{min}(b_q^h)$, for the lower and upper limits of interval $[g_j^{min}(b_q^h), g_j^{max}(b_q^h)]$ that will allow distinguishing the fuzzy indifference relation according to the rules described below. The DM should also be able to provide two veto thresholds, v_j^{min} and v_j^{max}, for the upper and lower limits of the interval $[g_j^{min}(b_q^h), g_j^{max}(b_q^h)]$ that would discriminate situations of discordance with the indifference relation.

The PROAFTN method requires the DM to provide the weight w_j^q for each criterion g_j. In PROAFTN, the meaning of the weight w_j^q of criterion g_j is specified, and it indicates the relative importance of the criterion when compared to other criteria in terms of votes that support the assignment in the class l_q.

The assignment procedure consists of calculating the degree of membership of each alternative to be assigned into each class based on the fuzzy indifference relation between this alternative and each reference alternative of each class. The alternative is assigned into a class with the maximal value of such a membership degree.

The indifference relation shows to which degree "one alternative is indifferent or roughly equivalent to another alternative". The indifference relation for the alternative x_i to be classified and the reference alternative b_q^h on the criterion g_j is defined according to the following rules:

1. If $g_j^{min}(b_q^h) \leq g_j(x_i) \leq g_j^{max}(b_q^h)$, then the alternative x_i to be classified is indifferent to the reference alternative b_q^h on the criterion g_j ($x_i I_j b_q^h$ is true).

2. If $g_j(x_i) \leq g_j^{min}(b_q^h) - d_j^-(b_q^h)$ or $g_j(x_i) \geq g_j^{max}(b_q^h) + d_j^+(b_q^h)$, then the alternative x_i to be classified is not indifferent to the reference alternative b_q^h on the criterion g_j ($x_i I_j b_q^h$ is not true).

3. If $g_j^{min}(b_q^h) - d_j^-(b_q^h) < g_j(x_i) < g_j^{min}(b_q^h)$ or $g_j^{max}(b_q^h) < g_j(x_i) < g_j^{max}(b_q^h) + d_j^+(b_q^h)$, then there is a weak preference between the alternative x_i to be

classified and the reference alternative b_q^h ($x_i Q_j b_q^h$ is true). In this case, the thresholds play the role of boundaries for the weak preference.

The proposed explanation of the relations is used for calculating the partial concordance indices $C_j^+(x_i, b_q^h)$ and $C_j^-(x_i, b_q^h)$ for each possible pair of the alternative x_i to be classified and the reference alternative b_q^h of the class l_q. Let us see how these indices are used in the PROAFTN method.

The algorithm of the PROAFTN method

Step 1 For each alternative x_i ($i = 1, \dots, m$) from the set of alternatives X to be classified and each reference alternative b_q^h ($h = 1, \dots, t_q$) of the class l_q ($q = 1, \dots, s$) from a set of possible ones L, compute the partial concordance indices (see Figure 11) on the criterion g_j ($j = 1, \dots, n$):

$$C_j^-(x_i, b_q^h) = \frac{d_j^-(b_q^h) - \min\{g_j^{min}(b_q^h) - g_j(x_i), d_j^-(b_q^h)\}}{d_j^-(b_q^h) - \min\{g_j^{min}(b_q^h) - g_j(x_i), 0\}}.$$

$$C_j^+(x_i, b_q^h) = \frac{d_j^+(b_q^h) - \min\{g_j(x_i) - g_j^{max}(b_q^h), d_j^+(b_q^h)\}}{d_j^+(b_q^h) - \min\{g_j(x_i) - g_j^{max}(b_q^h), 0\}}.$$

Then, set

$$C_j(x_i, b_q^h) = \min\{I_j^-(x_i, b_q^h), I_j^+(x_i, b_q^h)\}.$$

Step 2 Compute the partial discordance indices (see Figure 12):

$$D_j^-(x_i, b_q^h) = \frac{g_j(x_i) - \max\{g_j(x_i), g_j^{min}(b_q^h) - d_j^-(b_q^h)\}}{d_j^-(b_q^h) - \max\{g_j^{min}(b_q^h) - g_j(x_i), v_j^-(b_q^h)\}}.$$

$$D_j^+(x_i, b_q^h) = \frac{g_j(x_i) - \min\{g_j(x_i), g_j^{max}(b_q^h) + d_j^+(b_q^h)\}}{-d_j^+(b_q^h) + \max\{-g_j^{max}(b_q^h) + g_j(x_i), v_j^+(b_q^h)\}}.$$

Then, set

$$D_j(x_i, b_q^h) = \max\{D_j^-(x_i, b_q^h), D_j^+(x_i, b_q^h)\}.$$

Step 3 Calculate the fuzzy indifference relation $x_i I b_q^h$ with a indifference index:

$$I(x_i, b_q^h) = \sum_{j=1}^{n} w_j^q C_j(x_i, b_q^h)(\prod_{j=1}^{n} 1 - D_j(x_i, b_q^h))^{w_j^q},$$

where w_j^q are normalized and nonnegative weights (defined in formula (3)).

Step 4 Evaluate the fuzzy membership degrees $d(x_i, l_q)$. The membership degree is computed for each class from the set L by selecting the maximal values of indifference indices from the reference alternatives of each class:

$$d(x_i, l_q) = \max\{I(x_i, b_q^1), \dots, I(x_i, b_q^{t_q})\}.$$

Step 5 The alternative x_i is assigned to the class with a maximal membership degree:

$$x_i \in l_q \Longrightarrow d(x_i, l_q) = \max\{d(x_i, l_1), \dots, d(x_i, l_s)\}.$$

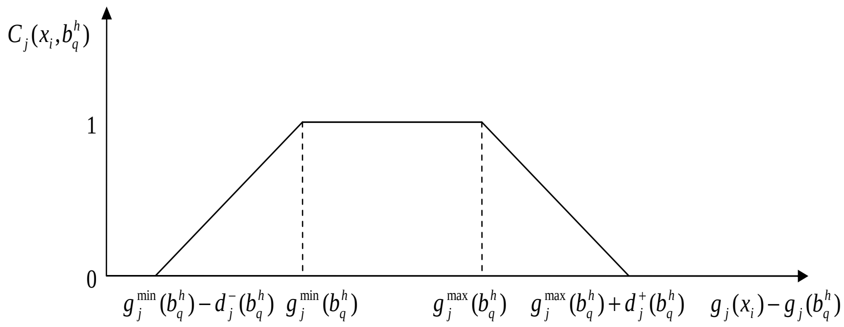

FIGURE 11 Concordance index for the PROAFTN method

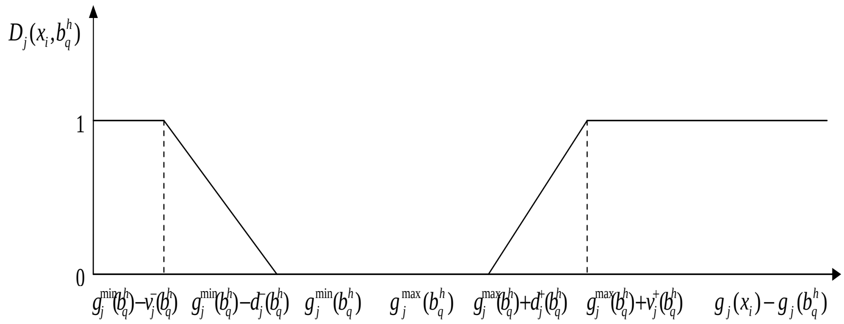

FIGURE 12 Discordance index for the PROAFTN method

In the same way as MC Filtering, PROAFTN is based on the concordance and non-discordance principles and have the advantages and disadvantages of this approach. Therefore, one of the advantages of PROAFTN is the flexibility when comparing pairs of alternatives (allowing incomparability and absence of need to have transitive relations). In addition, the developers define the reference alternatives in a fuzzy manner with the interval $[g_j^{min}(b_q^h), g_j^{max}(b_q^h)]$ of possible values on each criterion g_j. Such representation is more flexible than a definition of alternatives with crisp criteria values used in most of the methods and is suitable in situations where there are no possibilities to obtain exact reference alternatives. At the same time, such a representation substitutes the indifference threshold used in other outranking methods, as follows from $q_j(b_q^h) = g_j^{max}(b_q^h) - g_j(b_q^h) = g_j(b_q^h) - g_j^{min}(b_q^h)$.

However, there is still a lot of parametric information to be defined by the DM, including discrimination and veto thresholds as well as weights. The discrimination thresholds, d_j^+ and d_j^-, are very similar to the preference threshold $p_j(b_q^h)$: they both show the smallest difference between two alternatives such that one alternative is not considered to be indifferent anymore (in the case of the discrimination threshold) or is considered to be preferred (in the case of the preference threshold) to the other one on the criterion g_j. There are two discrimination thresholds used for two limits of interval of possible criteria values, consequently.

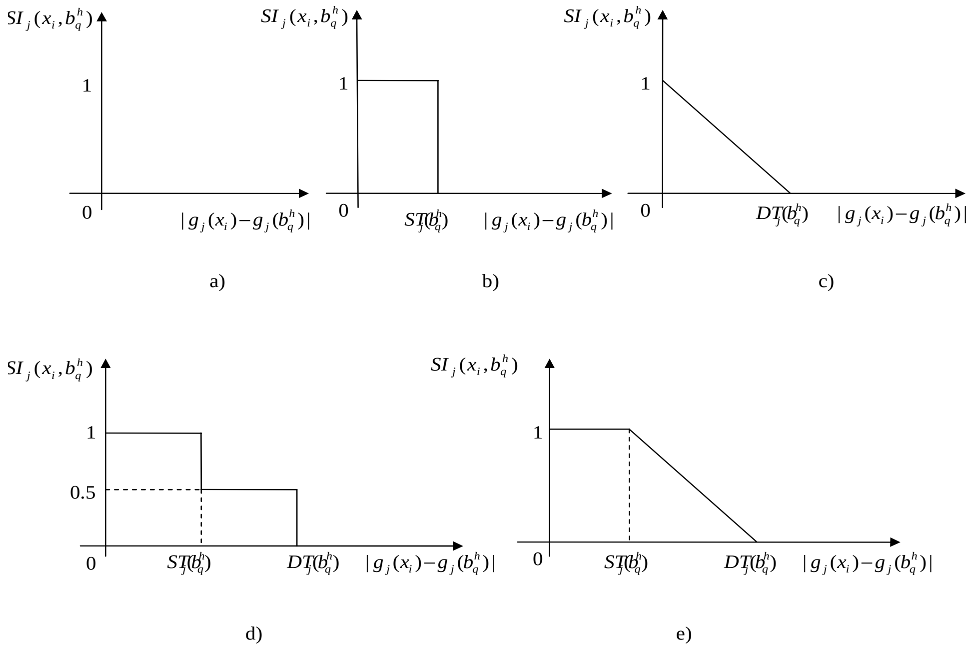

FIGURE 13 Similarity function for the TRINOMFC method

The weight of a criterion shows the relative importance of this criterion when compared to other criteria in terms of votes that support the assignment in a particular class. Such interpretation corresponds to nominal classification when compared to the treating of weights as votes that support or oppose the preference of one alternative over the other one in outranking approach for ordinal classification (as well as for choice and for ranking types of MCDA problems).

The use of preference relation calculated in nominal classification may be arguable. Here, there is no real need to define a preference situation between two alternatives but rather their similarity. It would be also good to have possibility to estimate the alternatives with the attributes and not only the criteria.

PROAFTN has been extensively applied to medical diagnostics, in particular to the cytopathological diagnosis of acute leukaemia [10], [11], [13]. There is also the PROCFTN method [12] developed using fuzzy indifference relation of PROAFTN for the selection of k reference alternatives for the k-nearest neighbors method [39]. The procedure for the preference elicitation of the PROCFTN method developed can be used to overcome the cognitive loading of the DM when specifying such parameters directly in [14].

3.9.2 TRINOMFC

The TRINOMFC method was developed by Léger and Martel [101] with the aim to reduce the amount of information needed for classification. For instance, with it one can avoid the necessity for defining a veto threshold (that may be a difficult task for the DM). In TRINOMFC, the DM has to define the set of alternatives X estimated on the set of criteria G to be classified in the set of classes L. There can be several reference alternatives for each class, and the set of reference alternatives is defined as $B = \{b_1^1, \ldots, b_1^{t_1}, \ldots, b_s^1, \ldots, b_s^{t_s}\}$. The DM is also asked to provide the weights of criteria that have the same meaning as in the PROAFTN method. Thus, the weight w_j^q of the criterion g_j shows the relative importance of the criterion when compared to other criteria for the assignment in the class l_q.

For the comparison of the alternative x_i to be classified and the reference alternative b_q^h of the class l_q, in this method, a function SI_j (represented by the similarity or indifference index $SI_j(x_i, b_q^h)$) is constructed on each criterion g_j. Such function varies on the interval $[0, 1]$ and has a greater value if the alternatives are more similar to each other on the particular criterion. The examples of types of functions for modeling similarity are presented in Figure 13 adapted from the criteria functions used in the PROMETHEE method [23]. However, any other type of function can be chosen arbitrary. The DM has also to define similarity and non-similarity thresholds, $ST_j(b_q^h)$ and $DT_j(b_q^h)$, associated to the reference alternative b_q^h for each criterion g_j. The steps of the methods can be represented as follows.

The algorithm of the TRINOMFC method

Step 1 At the beginning, the DM has to define the type of similarity function SI_j that would allow comparison of the alternative x_i $(i = 1, \ldots, m)$ to be classified and each reference alternative b_q^h $(h = 1, \ldots, t_q)$ of class l_q $(q = 1, \ldots, s)$ on each criterion g_j $(j = 1, \ldots, n)$, based on which the partial similarity index $SI_j(x_i, b_q^h)$ is constructed. For instance, if the DM selects the true-type similarity function depicted in Figure 13(a) for the criterion g_j, then the partial similarity index will look like:

$$SI_j(x_i, b_q^h) = \begin{cases} 1, \text{if } g_j(x_i) - g_j(b_q^h) = 0, \\ 0, \text{if } g_j(x_i) - g_j(b_q^h) \neq 0. \end{cases}$$

This function does not need threshold information to be provided by the DM and assumes that the two alternatives, x_i and b_j^h, are similar on the criterion g_j only if they have exactly the same values. For more complex than true-type functions, the DM has to define the similarity and dissimilarity thresholds, $ST_j(b_q^h)$ and $DT_j(b_q^h)$, respectively. In the same way as the indifference threshold $q_j(\cdot)$ in the ELECTRE TRI method (see Section 3.1.1), the similarity threshold $ST_j(b_q^h)$ indicates the largest difference between two alternatives on the criterion g_j such that they remain indifferent for the DM. The dissimilarity threshold $DT_j(b_q^h)$ is different from the preference threshold $p_j(\cdot)$ in the ELECTRE TRI method (see Section 3.1.1) because it indicates the smallest difference between the alternatives x_i and b_q^h such that the first alternative x_i is different (instead of being preferred in the preference threshold) from the second one b_q^h on the criterion g_j. For instance, for the pseudo-type similarity function in Figure 13(e) the partial similarity index on the criterion g_j will have the following form:

$$SI_j(x_i, b_q^h) = \begin{cases} 1, \text{if } |g_j(x_i) - g_j(b_q^h)| \leq ST_j(b_q^h), \\ 0, \text{if } |g_j(x_i) - g_j(b_q^h)| > DT_j(b_q^h), \\ \frac{DT_j(b_q^h) - |g_j(x_i) - g_j(b_q^h)|}{DT_j(b_q^h) - ST_j(b_q^h)}, \text{ otherwise,} \end{cases}$$

where $ST_j(b_q^h) > 0$, $DT_j(b_q^h) > 0$, and $ST_j(b_q^h) \neq DT_j(b_q^h)$.

Step 2 The next step is obtaining a global similarity index as an aggregation of the partial ones:

$$SI(x_i, b_q^h) = \sum_{j=1}^{n} w_j^h SI_j(x_i, b_q^h),$$

where w_j^h is the weight that shows the importance of the criterion g_j for the class l_q. The weights are normalized and nonnegative (defined in formula (3)).

Step 3 Among all the similarity indices calculated for the alternative x_i to be classified and each reference alternative b_q^h of the class l_q, the one for each class is selected based on the minimal principle or pessimistic rule. According to this rule, it is assumed that the "worst" of similarity index values is accepted:

$$d(x_i, l_q) = \min\{SI(x_i, b_q^1), \ldots, SI(x_i, b_q^{t_q})\}.$$

Step 4 As a result, an alternative is assigned to the class with the maximal value of the membership index between this alternative x_i and one of the reference alternatives b_q^h of this class l_q selected at the previous step:

$$x_i \in l_q \Longrightarrow d(x_i, l_q) = \max\{d(x_i, l_1), \ldots, d(x_i, l_s)\}.$$

Each alternative is assigned to only one class. If some alternative cannot be classified to exactly one class, then it is possible to define a cutting level λ:

$$d(x_i, l_q) \geq \lambda.$$

It is possible to use not only "min max" type of rule for the assignment of alternatives to be classified but also other ones, such as "max min" or Hurwicz assignment rules. For the case of several reference alternatives for each class, it is worth first checking the following rule: the similarity indices between the reference alternatives within each class should be smaller than the similarity indices between the reference alternatives from different classes. This rule should be checked before applying the TRINOMFC method.

The developers of TRINOMFC view nominal classification from the classical approach point of view. Indeed, they search for the similarities and differences between the alternative to be classified and the reference alternatives, unlike in the approaches that stem from the outranking methods, where the alternatives are compared based on the preference relation between them. They have also searched a way to reduce the amount of information provided by the DM and have managed to avoid the necessity for defining a veto threshold.

On the other hand, instead of preference and indifference thresholds in TRI-NOMFC similarity and dissimilarity thresholds are used. The similarity threshold has similar meaning to the indifference threshold in the outranking theory and shows maximal difference on the criterion value that is still small enough for two alternatives to be considered similar. The dissimilarity threshold represents a minimal value of difference between two values on the criterion that makes one alternative to be dissimilar to another one (when compared to the preference threshold that shows the preference of one alternative over another one). Thus, such thresholds are more natural for nominal classification problems.

The use of different types of similarity functions allows flexibility when comparing two alternatives. In the method, the alternatives are estimated on the criteria only; however, the method can be easily adapted to the use of attributes. In this way, the true-type similarity function can be applied. Two alternatives are considered to be similar on the same attribute if they have exactly the same values on this attribute.

Now that the most widely-spread MCDA methods for classification have been introduced, we have a background for focusing on the development of new methods by taking into consideration the advantages of different methods while trying to reduce their disadvantages.

4 NEW CLASSIFICATION METHODS

As has been mentioned above, in real-life situations, for the DM it may be difficult to think in terms of parameters of a particular model or define the exact numerical values for parameters. In this work, we concentrate on methods that avoid direct interrogation of parameter values from the DM. According to the categorization of methods provided in the introduction to Chapter 3, such methods belong to the indirect parameter-based and parameter-free groups of methods. Most of the methods from these groups use assignment examples in order to perform classification. The definition of assignment examples (if they are available) is usually much easier than the definition of the exact parameter values for the DM. The survey of indirect parameter-based methods shows that there are two groups of methods based either on mathematical programming or simulation, and most of the methods belong to the first group. However, solving of mathematical programming problem can be challenging, especially, for nonlinear problems that may require computational time and a powerful computer. That is why, in this work, we have developed a simulation-based method for nominal classification, the so-called SMAA-Classification method. The method assumes that the set of alternative to be classified is estimated on a set of criteria and/or attributes with numerical scales of values. Furthermore, the number of classes and assignment examples for each of them should be available a priori, to be defined by the DM for instance.

When compared to SMAA-TRI for ordinal classification that uses ELECTRE TRI with uncertainty or imprecise parameters, SMAA-Classification allows nominal classification based on assignment examples. SMAA-Classification does not require any parametrical information to be given in advance by the DM. However, if such information is available (for instance, in terms of intervals or distributions of weights or criteria values), it can be used. In a similar way as it is done in the SMAA-TRI method presented in Section 3.4, the SMAA-Classification method provides the DM with descriptive information about the probability of each alternative to be assigned into each class as well as preferences (e.g., in the form of weights) that support such assignment. Even though the SMAA-Classification method does not require any parameter information to be defined in advance, it performs as efficiently as methods that assume the parameters of the model to be defined by the DM (for details, see Section 6.1).

On the other hand, there can be situations, where it is not possible to provide assignment examples. The survey of the existing MCDA methods for classification presented in Chapter 3 shows that, currently, VDA methods introduced in Section 3.5 include procedures of interaction with the DM and provide the tools for questioning the DM about his or her preferences. These methods (e.g., SAC and ORCLASS) have been developed in such a way that the DM has to classify some alternatives, and the rest of the classification is done by the method. They do not need any other information to be available in advance except the number of classes and criteria with the scales of values (the SAC method also assumes the set of given alternatives to be provided). The survey of VDA methods for classification presented in Section 3.5 shows that there is still space for improvement in the existing methods. That is why we have developed a new Dichotomic Classification method that allows classification of the set of given alternatives described by verbal or numerical scales of criteria values. The Dichotomic Classification method follows VDA principles: it is interactive and it presents some alternatives from the set of given alternatives to the DM for direct classification, while the rest of the alternatives are classified by the method. The alternatives estimated with verbal values are presented to the DM in a natural language. In a similar way with the VDA methods developed earlier, the Dichotomic Classification method allows classification of each alternative into exactly one class. When compared to the other VDA methods, Dichotomic Classification is more effective on the same data sets in terms of number of alternatives posed to the DM for direct classification. It is also more efficient computationally and does not depend on the distribution of alternatives between classes (for details, see Section 6.2). Next, we present and discuss these two new methods in more detail.

4.1 SMAA-Classification

In what follows, we introduce the so-called SMAA-Classification method for nominal classification. The method assumes that the set of alternatives to be classified is available. These alternatives can be estimated on a set of criteria and/or attributes that have numerical scales of possible values. The method assists in the assignment of alternatives when no preferences of DM are available or when they are imprecise. There can also be preferences of several DMs to be taken into account. The SMAA-Classification method considers them in the same way as imprecise preferences of a single DM and finds some average preferences of a typical DM.

The SMAA-Classification method based on the Monte-Carlo simulation calculates at each iteration the distances between the alternatives to be classified and the assignment examples that are assumed to be available. At each iteration, the method assigns an alternative into a class if the distance to the assignment example is minimal when compared to the distances to the assignment examples of the other classes. Based on the analysis of statistical information obtained at the end of the simulation process, the method calculates the levels of acceptability of the same alternative into different classes. If additional information is needed, the method can also define the central weight vectors with which such assignments are supported. This descriptive information is provided to the DM for further evaluation. There is also the possibility of the criteria values to be imprecise. In case the classification to exactly one class is required, the DM is asked to define a value of the acceptability index that is sufficient for classification of each alternative to only one class. In such a way, the ambiguous cases with high levels of acceptability for several classes may be resolved.

4.1.1 Introduction

We develop a method for nominal classification problems in situations where there is no "exact" information about reference alternatives, but where there is at least one assignment example for each class. These assignment examples may be a set of decisions that the DM used making in the past; or decisions made for realistic alternatives not considered in the set of given alternatives; or decisions for a limited set of alternatives from the set of given alternatives [72]. This case is typical for a real-world situation where the DM usually has some examples of classification (at least one) for each class. Defining reference alternatives may be a more difficult task. For instance, in medical diagnostics doctors feel confident about clear cases with known symptoms for a certain disease but hesitate when making decisions about assigning cases that are not clear or that have symptoms of several diseases.

SMAA-Classification for nominal classification [164] is an MCDA method that allows assigning a set of given alternatives evaluated on a set of criteria and/or attributes into a set of classes predefined with assignment examples and considers the preference information to be imprecise or uncertain.

At the beginning, the DM specifies all initial data: the set of alternatives $X = \{x_1, \dots, x_m\}$ estimated on the set criteria and/or attributes $G = \{g_1, \dots, g_n\}$ and to be classified to the set of classes $L = \{l_1, \dots, l_s\}$, and the set of assignment examples $B = \{b_1^1, \dots, b_1^{t_1}, \dots, b_s^1, \dots, b_s^{t_s}\}$. For each criterion and/or attribute g_j from the set G, we have the scale of values $S(g_j) = \{g_{j1}, \dots, g_{jt}\}$ (where t is the number of values on the scale of criterion and/or attribute g_j). The values on the scale of each criterion are ordered. The most desirable value has the maximal rank.

We assume that we have at least one assignment example for each class. The SMAA method allows that the criteria and/or attributes values as well as DM's preferences that support assignment to be imprecise or uncertain. That is why the criteria and/or attribute values of the alternative x_i can be stochastic. Then, we can use the notation ξ_i for defining the alternative x_i estimated with the stochastic values $g_j(\xi_i)$ on each criterion and/ot attribute g_j. The stochastic values can be presented by a joint probability distribution $f(\xi)$ in the space $R^{m \times n}$, which can be available as a partial knowledge about criteria and/or attribute evaluations but, if not specified, is assumed to be uniform.

We can use the following ideas in the classification: assignment of an alternative 1) according to the maximal similarity to at least one of the assignment examples of the class, or 2) with regards to the minimal distance between the alternative to be classified and at least one of the assignment examples of the class. In most of the recently developed MCDA classification methods [10], [110], [101], [131], an approach based on the estimation of some similarity index is used. The similarity indices are different in different MCDA methods. For instance, in the TRINOMFC method [101] (see Section 3.9.2), an alternative is assigned into a class based on the similarity index known as the "indifference index". Another approach to model similarity has been used in clustering analysis [64]. In this approach, an alternative is assigned into a class if the distance to the assignment example of this class is minimal when compared to the distances to the assignment examples of other classes.

The distance can be calculated with different metrics such as the Euclidean distance (L_2-norm distance), the Mahalanobis distance (that is an Euclidean distance that takes into account correlations of the data set), the Manhattan distance (L_1-norm distance), the Minkowski distance (L_p-norm distance) and others [39]. The most general type of the metric is the Minkowski distance, according to which the distance between an alternative x_i to be classified and an assignment example b_q^h of the class l_q is:

$$d^M(x_i, b_q^h) = (\sum_{j=1}^{n}(|g_j(x_i) - g_j(b_q^h)|)^p)^{1/p}, \tag{18}$$

where $g_j(x_i)$ and $g_j(b_q^h)$ are the values of the alternatives x_i and b_q^h on the criterion g_j, respectively, and the value of the $p \geq 1$ parameter is selectable. Thus, with $p = 1$ the Minkowski distance converts to the Manhattan distance, and with $p = 2$ it becomes the Euclidean one.

We assign the alternative x_i into the class l_q if the distance to the assignment example b_q^h of this class is smallest (at iteration k) when compared to the distances to the assignment example of the other classes:

$$x_i \in l_q \Longrightarrow q = class(x_i^k) = \arg \min_{\substack{1 \leq q \leq s \\ h=1...t_q}} d^M(x_i, b_q^h),$$

$$x_i \in l_q \Longrightarrow q = class(x_i^k) = \arg \min_{\substack{1 \leq q \leq s \\ h=1...t_q}} \left\{ (\sum_{j=1}^{n}(|g_j(x_i) - g_j(b_q^h)|)^p)^{1/p} \right\}.$$

For the case where there is information about the relative importance of the criterion values being close to assignment example values in the form of weights, we can use the weighted Minkowski distance:

$$d^{WM}(x_i, b_q^h) = (\sum_{j=1}^{n}(w_j|g_j(x_i) - g_j(b_q^h)|)^p)^{1/p}.$$

In the present work, for the estimation of the distance between alternatives we select the weighted Euclidean distance as it is the most commonly used one:

$$x_i \in l_q \Longrightarrow q = class(x_i^k, w^k) = \arg \min_{\substack{1 \leq q \leq s \\ h=1 \ldots t_q}} d^{WE}(x_i, w, b_q^h), \tag{19}$$

$$x_i \in l_q \Longrightarrow q = class(x_i^k, w^k) = \arg \min_{\substack{1 \leq q \leq s \\ h=1 \ldots t_q}} \left\{ \sqrt{\sum_{j=1}^{n} (w_j |g_j(x_i) - g_j(b_q^h)|)^2} \right\}.$$

The SMAA methodology (see Section 3.4) explores the space of imprecise or uncertain parameters in order to find such of them that support each scenario. For classification problem, such a scenario is the assignment of an alternative to a class. Thus, the SMAA-Classification method searches for such parameter values that classify an alternative into a given class. For instance, in the weighted Euclidean distance one may be interested in exploring such parameters as weights.

According to the SMAA-Classification method, after the distance function is selected, the Monte-Carlo simulation is organized for each alternative to be classified in the following way. Random sets of weights are generated from the nonnegative and normalized weight space, $w \in W$, defined in the same way as in formula (3).

Then, with regards to the selected distance function using a given weight vector, the alternative to be classified is assigned into one of the classes according to formula (19). This procedure is repeated for the number of simulation iterations selected. The number of iterations is selected with regards to the accuracy requirements. On the results in [152], it is shown that 10000 iterations is typically enough for a reliable simulation with SMAA. As a result of the Monte-Carlo simulation, statistical information for each alternative to be assigned into each class is collected. This information includes the number of successful classifications into each class and aggregated preferences that have been simulated for such successful classifications. Then, the SMAA-Classification method may calculate descriptive information in the form of acceptability index for each alternative to be classified into each class. The relation between the number of successful classifications and the total number of iterations yields an acceptability index. Next, we introduce descriptive indices for SMAA-Classification by analogy with other SMAA methods; however, we represent them as a discrete measures and not as a continuous ones as it is usually done (see Section 3.4).

An *acceptability index* shows the variety of different successful valuations that assign the alternative x_i into the class l_q:

$$a_i^q = \frac{1}{z} \sum_{k: class(x_i^k, w^k) = q} 1,$$

where z is total number of simulation runs and k is a successful iteration at which the alternative x_i is assigned into the class l_q and w^k is the weight vector with which such successful classification has been obtained.

The higher the value of the acceptability index $a_i^q \in [0,1]$ the higher the probability for the alternative x_i to be assigned into the class l_q. Thus, $a_i^q = 0$ indicates that the alternative x_i is never assigned into the class l_q, and $a_i^q = 1$ shows that the alternative x_i is assigned into the class l_q with any set of simulated weights.

If additional information is desired in SMAA-Classification, we can give information about favorable weight vectors with which successful classifications have been obtained. The favorable weight vector shows the importance of each criterion in the distance function for classification of the alternative into the class in the following way. If it is important that the criterion value of an alternative to be classified is close to the corresponding assignment example value, then the weight should be big.

Thus, from all simulated weight vectors the favorable ones are collected in $W^q(x_i)$ with which the alternative x_i is successfully assigned into the class l_q:

$$W^q(x_i) = \{w^k \in W : class(x_i, w^k) = q\}. \tag{20}$$

Then, a central weight vector may be calculated by averaging the set of favorable weights. The central weight vector $w_i^q = (w_{i1}^q, \ldots, w_{in}^q)$ represents averaged weights that support classification of the alternative x_i into the class l_q. Each component w_{ij}^q $(j = 1, \ldots, n)$ of the vector w_i^q is calculated as:

$$w_{ij}^q = \sum_{k:class(x_i^k, w_j^k)=q} 1 \Bigg/ \sum_{k:class(x_i^k, w_j^k)=q} \frac{1}{w_j^k}, \tag{21}$$

where k is the successful iteration at which the alternative x_i is assigned into the class l_q and w_j^k is the component of the simulated weight vector (on the criterion g_j) with which such successful classification has been obtained.

This may be calculated as the harmonic average of all the weights that provide a successful assignment of the alternative x_i into the class l_q. The descriptive information (acceptability indices and central weight vectors) is provided to the DM for further analysis.

The algorithm of the SMAA-Classification method

Step 1 Simulate weights that are assumed to be stochastic according to uniform distribution, if not specified otherwise.

Step 2 Perform classification of the set of given alternatives with regards to the selected distance function using a simulated weight vector. Save all the results necessary for calculating SMAA indices.

Step 3 Compute the descriptive information in the form of SMAA-Classification acceptability indices according to formula (20) and central weight vectors according to formula (21).

Step 4 Provide the DM with the SMAA indices. If the DM is not satisfied with the obtained classification, suggest him or her to rethink the initial information and specify it. Otherwise, if the DM agrees with results of the classification, the procedure is finished.

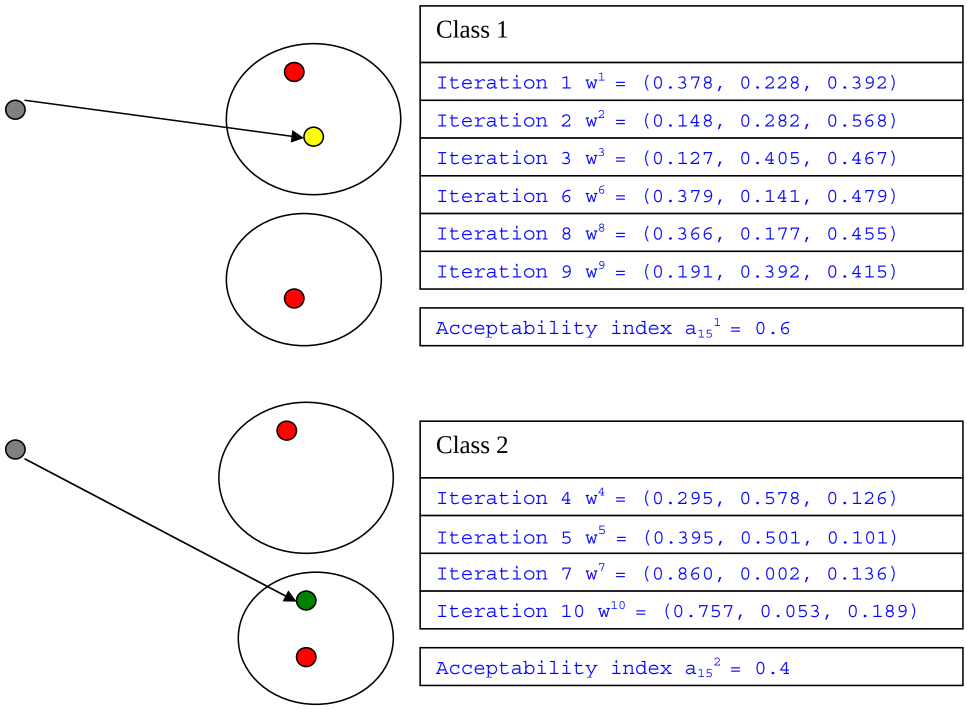

FIGURE 14 Simulation with 10 iterations for alternative $x_{15} = (0.2, 0.2, 0.3)$

Next, we illustrate the performance of the SMAA-Classification method with a simple example. Even though the method allows criteria and/or attribute values to be imprecise, in the illustrative example and the applications presented they are exact.

4.1.2 An Illustrative Example

In order to demonstrate the SMAA-Classification method, we consider an illustrative example with an artificial set of 27 alternatives evaluated on 3 criteria. This set is formed as a Cartesian product of criteria values $\{0.1, 0.2, 0.3\}$. The alternatives are to be classified into two classes that both have one assignment example: the alternative $b_1^1 = x_6 = (0.1, 0.2, 0.3)$ is the assignment example for the first class l_1, and the alternative $b_1^2 = x_{13} = (0.2, 0.2, 0.1)$ is the assignment example for the second class l_2. We run the SMAA-Classification method with 10 and 10000 iterations in the Monte-Carlo simulation.

As an example of how the calculation is realized, we show the results of a simulation with 10 iterations for the alternative $x_{15} = (0.2, 0.2, 0.3)$ presented in Figure 14. Let us follow the evaluation of this alternative at each step of the SMAA-Classification method.

At the beginning, the method simulates random weights with regards to some distribution, for instance, uniform, in such a way that $\sum_{j=1}^{n} w_j = 1$. At the

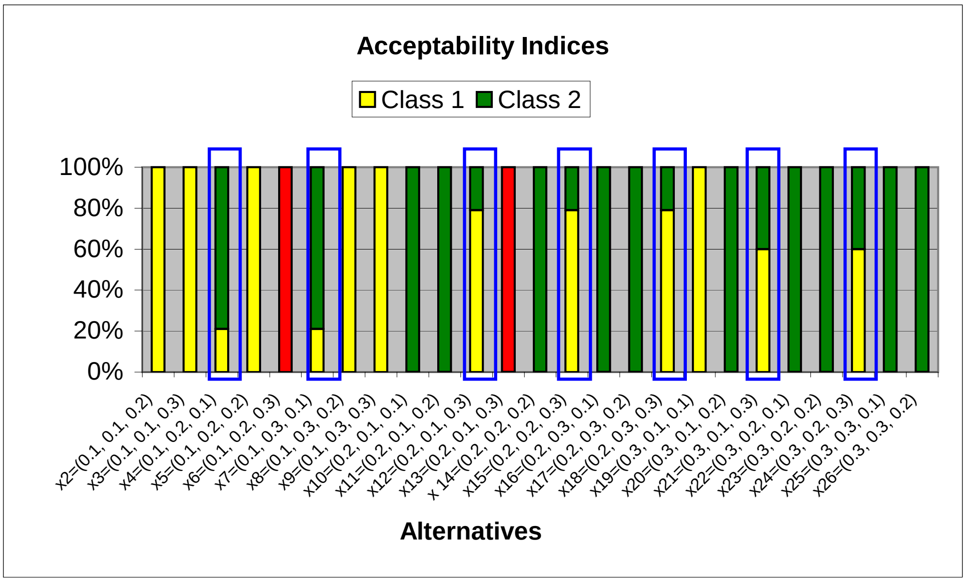

FIGURE 15 Diagram of alternatives with acceptability indices

first iteration, we obtain the weights $w^1 = (0.378,\ 0.228,\ 0.392)$. (Note that the sum of weights is not equal to one with the precision used for the reason that we only show three decimals here.) Then, the distances between the alternative to be classified x_{15} and the assignment examples x_6 and x_{13} are calculated. The alternative x_{15} is assigned into the class l_1 according to the minimal distance. Then, these two steps are repeated as many times as there are iterations. After 10 iterations, we have the following situation: with different simulated weights the alternative x_{15} appears 6 times in the class l_1 and 4 times in the class l_2, which corresponds to the acceptability indices $a^1_{15} = 0.6$ and $a^2_{15} = 0.4$ for the two classes. Then, the central weight vectors for each class can be calculated by averaging the random weights, with which the alternative x_{15} has been assigned into each class. Thus, we can obtain the central weight vectors $w^1_{15} = (0.264, 0.270, 0.462)$ and $w^2_{15} = (0.576, 0.283, 0.138)$ for the classes l_1 and l_2, respectively.

Usually, 10 iterations is not enough to get a complete picture of all possible random situations. That is why we have run the same example with 10000 iterations. The results are presented in Figure 15 in the form of a diagram with the acceptability indices calculated for our 27 alternatives. Even though here we consider the set of given alternatives to be equal to the Cartesian product of criteria scales, it should not be necessary the case and we may perform the classification of arbitrary given set of alternatives. Here, each bar shows the percentage of acceptability for assigning the alternative into each class. For instance, from the very first bar on the left we can see that the alternative $x_2 = (0.1, 0.1, 0.2)$ is assigned into the class l_1 with 100% of random weight vectors. This means that the alternative x_2 is assigned into the class l_1 with any simulated weight vector. A similar situation appears with the alternatives x_3, x_5, x_8, x_9. For the alternatives

x_{10}, x_{11}, x_{14}, x_{16}, x_{17}, x_{20}, x_{22}, x_{23}, x_{25}, x_{26}, there is an unambiguous classification into the class l_2. For the rest of the alternatives, the assignment is not so obvious. For instance, the alternative $x_4 = (0.1, 0.2, 0.1)$ is assigned into the class l_1 with 21% of simulated weight vectors and into the class l_2 with 79% of cases.

Table 3 presents the results of classification for the alternatives with the acceptability indices different from 0 or 1. The DM may be interested to see the classes where each alternative may be classified. It is up to the DM to define the value of the acceptability index that is sufficient for crisp assignment of an alternative into a class. Alternatively, the DM may pay more attention to such alternatives.

TABLE 3 Alternatives with acceptability indices different from 0 and 1

Alternative	Class 1	Class 2
x_4=(0.1, 0.2, 0.1)	a_4^1 = 0.21	a_4^2 = 0.79
x_7=(0.1, 0.3, 0.1)	a_7^1 = 0.21	a_7^2 = 0.79
x_{12}=(0.2, 0.1, 0.3)	a_{12}^1 = 0.79	a_{12}^2 = 0.21
x_{15}=(0.2, 0.2, 0.3)	a_{15}^1 = 0.79	a_{15}^2 = 0.21
x_{18}=(0.2, 0.3, 0.3)	a_{18}^1 = 0.79	a_{18}^2 = 0.21
x_{21}=(0.3, 0.1, 0.3)	a_{21}^1 = 0.6	a_{21}^2 = 0.4
x_{24}=(0.3, 0.2, 0.3)	a_{24}^1 = 0.6	a_{24}^2 = 0.4

4.1.3 Concluding Remarks and Future Research

The advantage of the SMAA-Classification method for nominal classification is an absence of a need for any parametric information to be defined by the DM. Instead, the assignment examples that can be expected to be easier to define. On the other hand, SMAA-Classification can also be used as a tool for modeling and solving classification tasks in situations where imprecise or uncertain information about parameter values is available.

When compared to the most developed methods that require either assigning the alternative to exactly one class (and resolve any ambiguous classification) or show the range of admissible classes without providing probabilities of assignment into each class, SMAA-Classification provides a general picture with a level of acceptability for each alternative to be assigned into each class. In addition, the SMAA-Classification method provides information about typical preferences that support each assignment. The method allows the alternatives to be estimated on attributes and/or criteria.

The performance of SMAA-Classification depends on the set of assignment examples, its quality and number of examples for each class. Indeed, the closer this set is to the set of reference alternatives (that are the brightest representatives of classes), the better the resulting classification. Even though the quality of the set of assignment examples is very important, we also have to study in future the influence of the size of this set on the quality of classification.

The SMAA-Classification method can also be extended to the case where the assignment examples are absent. Then, some clustering techniques can be applied. This is a topic of future research. Another issue to be investigated is the usage of the metrics. Currently, we have applied the weighted Euclidean metrics for the calculation of the distance between two alternatives. However, other metrics could be studied for different situations. Another interesting matter to be investigated is the validity of the aggregation of the criteria with the distance function. When compared to the attributes that are neither minimized, nor maximized, the criteria scales have the direction of preferences that should be considered in the aggregation function; thus, all criteria should be reduced to the same direction of preference. We are also planning to study the possibility of taking the PROMETHEE criteria functions into account in SMAA-Classification in a similar way as in TRINOMFC.

In the future, it would also be interesting to test, for the SMAA-Classification method, the possibilities of interactive improvement of the set of assignment examples in a similar way as was done for the ELECTRE TRI method in [32]. Another interesting question concerns resolving contradictions among assignment examples that may appear, for instance, if several DMs assign the same alternative to different classes or if the DM is inconsistent. Similar work has been done with ELECTRE TRI in [115].

Currently, the SMAA-Classification method need the set of assignment examples to be available a priori. But, in some real-life situation, it may be difficult for the DM to define assignment examples. That is why the next method presented has been developed with no need for such a set to be available in advance.

4.2 Dichotomic Classification

Here, we introduce the so-called Dichotomic Classification method for ordinal classification that has been developed in the framework of VDA. The method assists in the assignment of alternatives when no preferences of the DM are available in advance. It is parameter-free method, and it does not require the DM to think in terms of any model parameters. There can be criteria with verbal or numerical scales of values. The method is interactive, and at each iteration some alternative from the set of given alternatives is posed to the DM for direct classification. Thus, some alternatives are assigned by the DM and the rest of the alternatives are classified by the method. The method assigns each alternative to exactly one class.

When compared to the VDA methods presented in Section 3.5.1, the Dichotomic Classification method can work for the set of given alternatives without using the complete set obtained as a Cartesian product (that is an universal set of alternatives for the selected set of criteria) of all criteria scales. However, the DM should be questioned for the classification of each such new set of given alternatives. The exception is the case when the set of given alternatives is equal to the complete set of alternatives obtained as a Cartesian product of all criteria values. Then, we can search for the rules that describe classes with boundary alternatives in a similar way it is done in the ORCLASS and SAC methods (see Section 3.5.1). Then, there is no need of questioning the DM each time the new set of alternatives (that is smaller than the Cartesian product of criteria scales) has to be classified. A boundary alternative can be detected when all the alternatives have been classified with regards to the property, according to which, for the boundary alternative, changing the value on at least one criterion moves it to a neighboring class.

4.2.1 Introduction

While developing the new method we aim at improving the efficiency of the VDA classification procedure (presented in Section 3.5.1): to increase the number of alternatives that can be classified without increasing the computation time and to decrease the number of alternatives proposed to the DM for direct classification. This can be done by simplifying the rule of selecting the alternative to be given to the DM for classification at each iteration of the preference elicitation procedure. In the ideal case, the method should effectively work for different alternative distributions between classes.

We assume that criteria can have verbal estimations on the criteria values and that information about the importance of the criteria can be absent. On the other hand, sometimes information about a lexicographic ordering of criteria is available. The lexicographic ordering [108] assumes that a more important criterion is infinitely more important than a less important criterion. We consider using lexicographic ordering of criteria when we have enough information for it. Thus, in the method two cases are distinguished, i.e., when the criteria are lexicographically ordered in terms of importance for the DM and when they are not. As before, we are solving a maximization problem, where higher values are preferred.

The idea of the method proposed follows the principles of the VDA methods for ordinal classification (see Section 3.5.1): to realize the classification of a set of given alternatives (that may have verbal values on the scales of criteria) with a minimal number of classifications made by the DM. The method is called Dichotomic Classification [163] and it is based on two concepts: VDA [100] and the bisection method or a modification of a more popular dichotomous search [149].

The initial information needed for the Dichotomic Classification method is the same as for SAC: the number of classes and their order, the set of criteria with the scale of values for each of them as well as the order of criteria (if it is available) and the set of given alternatives. We consider the set of given alternatives to be equal to the set of alternatives obtained as the Cartesian product of scales of criteria, as in ORCLASS. However, the method is also able to work on the set of alternatives given by the DM.

The Dichotomic Classification method can be presented as follows. First, the DM specifies all necessary initial data: the set of criteria $G = \{g_1, \ldots, g_n\}$ and the set of classes $L = \{l_1, \ldots, l_s\}$. For each criterion g_j from the set G, we have the scale of values $S(g_j) = \{g_{j1}, \ldots, g_{jt}\}$ with the following ranks $R(g_j) = \{1, \ldots, t\}$ for them (where t is the number of values on the scale of criterion g_j). The set of alternatives is defined as the Cartesian product of all scales of criteria $X = S(g_1) \times S(g_2) \times \cdots \times S(g_n)$. The values on the scale of each criterion are ordered. The most desirable value has the maximal rank due to the nature of criterion. The classes are also ordered. The alternatives that belong to the class with the maximal rank are the most desirable ones. Two cases are distinguished, i.e., when the criteria are lexicographically ordered in terms of importance for the DM and when they are not.

The set of possible classes L_i is defined for each alternative x_i, where $x_i \in X$ ($i = 1, \ldots, m$). In the beginning, this set contains all the possible classes for each alternative, that is, $L_i = L$. However, with the classification of some alternatives the number of possible classes is reduced (according to the rule described below) until for each alternative it becomes equal to one, e.g., $|L_i| = 1$. Then, the classification is finished.

In ORCLASS and SAC, the selection of the alternative, which is proposed to the DM for classification, is based on its informativeness. The most informative alternative should classify, indirectly, a maximal number of other alternatives. The informativeness for each alternative is recalculated at each iteration. This task is complex and time-consuming for large sets of alternatives. We propose a much simpler way to select the alternative to be shown to the DM for classification using the bisection of the set of given alternatives at each iteration.

In verbal analysis, the qualitative values on the criteria, which are not applicable for mathematical operations such as addition and subtractions, are compared. However, it is possible to operate with ranks of alternatives on the scales of criteria. For comparison of verbal values, we should assign the ranks to the values on the scales of criteria according to their order. As mentioned, for the criterion g_j the scale of values $S(g_j)$ is with the ranks $R(g_j)$.

According to the bisection method, for defining the "middle" value of the alternative x_{middle} on the scale of criterion g_j, it is necessary to calculate the rank for it. For instance, at the first iteration, the rank of the "middle" alternative on the scale of criterion g_j is calculated as follows: $r_j(x_{middle}) = \frac{r_j(x_{best}) + r_j(x_{worst})}{2}$, where $j = 1, \ldots, n$, $r_j(x_{best})$ and $r_j(x_{worst})$ are ranks of the "best" and the "worst" alternatives for each criterion. If we have an even number of values on the scale of criterion, the value of the middle rank $r_j(x_{middle})$ will not be an integer. In this case, it is possible to round it either to the smaller or the bigger closest integer. In the method, we round it to the smaller one. Then, the value located at the rank $r_j(x_{middle})$ is considered as the value of the "middle" alternative on the scale of criterion g_j.

The principles of VDA assume that the preference relation works on the set of alternatives and that the relation of order on the set of classes is utilized. We also check how SAC and Dichotomic Classification work with all criteria being of equal importance to the DM and with a lexicographic order on the set of criteria. The Dichotomic Classification method utilizes two basic relations of the ORCLASS method: the relation of preference between the alternatives and the relation of order of classes (see Section 3.5.1).

In this work, we also consider the possibility of establishing (when it is possible) a lexicographic order on the set of criteria according to the importance of each criterion to the DM:

$$g_j \text{ P } g_f (\text{the criterion } g_j \text{ is infinitely more important than the criterion } g_f),$$
$$\text{if } j > f \text{ and } j, f = 1, \ldots, n; j \neq f.$$

In general, the sequence of steps in the algorithm of the Dichotomic Classification method is similar to ORCLASS even though it differs in details. Let us consider the algorithm of the Dichotomic Classification method.

The algorithm of the Dichotomic Classification method

Step 1 At the first stage, the initial data are defined: the alternatives "best" x_{best} (the alternative that has the most desirable value on each criterion) and "worst" x_{wost} (the alternative that has the least desirable value on each criterion) from the set of alternatives X (defined as the Cartesian product of the scales of criteria) are classified to the classes with the maximal and the minimal ranks, respectively. For each alternative x_i ($i = 1, \ldots, m$) from the set X, the set of possible classes L_i is assigned. At the beginning, this set contains all classes $L_i^{old} = L_i = L$.

Step 2 The stopping criterion for the classification is the absence of alternatives in the set of given alternatives with more than one possible class. Due to this, each alternative is classified to exactly one class. If this is not the case, a new iteration of the classification procedure starts.

Step 3 The "middle" alternative x_{middle} is calculated as an alternative that has middle ranks on scales of criteria with regards to the ranks of the "best" x_{best} and the "worst" x_{wost} alternative values. The "middle" alternative x_{middle} defines the boundaries for the two searching sets in the space of alternatives: one is the set of all alternatives between the "best" x_{best} and the "middle" x_{middle} alternatives, and another one is the set of all alternatives between the "middle" x_{middle} and the "worst" x_{wost} alternatives. In the following iterations, the "middle" alternatives are determined correspondingly.

Step 4 The "middle" alternative x_{middle} is proposed to the DM for classification. As a result, we obtain the class, to which he or she assigns the alternative x_{middle}, for instance, to the class l_q ($q = 1, \ldots, s$): $x_{middle} \in l_q$.

Step 5 At this point, the existence of inconsistencies between the current and previously made classifications is checked. The conditions related to previous classifications are checked:

$$\text{if } x_{middle} \in l_k \text{ and } x_{middle} \text{ P } x_i, \text{ then } x_i \in l_q \text{ where } q > k,$$
$$\text{if } x_{middle} \in l_k \text{ and } x_i \text{ P } x_{middle}, \text{ then } x_i \in l_q \text{ where } q < k.$$

Inconsistencies are checked in case of more than two classes. For two classes, there are no inconsistencies because once the DM has assigned an alternative to a class, all other alternatives with not less preferable values upon all criteria will be assigned to the same class. Inconsistencies are eliminated by asking the DM to think over the assignment of contradicting alternatives.

Step 6 According to the class for the alternative obtained from the DM at *Step 4*, the possible classes for each not yet classified alternative are recalculated:

$$\begin{aligned}
&\text{if } x_{middle} \in l_k \text{ and } x_{middle} \text{ P } x_i, \text{ then } L_i = L_i^{old} \cap l_1, \ldots, l_{k-1},\\
&\text{if } x_{middle} \in l_k \text{ and } x_i \text{ P } x_{middle}, \text{ then } L_i = L_i^{old} \cap l_k, \ldots, l_s,\\
&L_i^{old} = L_i.
\end{aligned}$$

As a result, for each alternative x_i which has not yet been classified, the set of possible classes L_i is determined. If the alternative has only one possible class, e.g., $|L_i| = 1$, it is classified indirectly. At the end of this step, the iteration is finished and the new one should be started at *Step 2*.

In the situation where a lexicographic order of the criteria is available, it is possible to order all the alternatives before the classification starts and each alternative will have its rank, for instance, r_i would be the rank of the alternative x_i. The procedure of classification is then simplified and *Steps 3, 5* and *6* are modified in the following way.

At *Step 3*, the middle alternative is selected with regards to the ranks r_{best} and r_{worst} of the "best" x_{best} and the "worst" r_{worst} alternatives that defines the first searching set: $r_{middle} = \frac{r_{best}+r_{worst}}{2}$. The rank r_{middle} of the middle alternative is rounded if necessary (when the obtained rank is not an integer) to the smaller closest integer, for instance.

At *Step 5*, the existence of inconsistencies is checked with regards to the conditions:

$$\begin{aligned}
&\text{if } x_{middle} \in l_k \text{ and } r_{middle} > r_i, \text{ then } x_i \in l_q \text{ where } q > k,\\
&\text{if } x_{middle} \in l_k \text{ and } r_i > r_{middle}, \text{ then } x_i \in l_q \text{ where } q < k.
\end{aligned} \tag{22}$$

The recalculation of the set of classes for each alternative at *Step 6* is done in the following way:

$$\begin{aligned}
&\text{if } x_{middle} \in l_k \text{ and } r_{middle} > r_i, \text{ then } L_i = L_i^{old} \cap l_1, \ldots, l_{k-1},\\
&\text{if } x_{middle} \in l_k \text{ and } r_i > r_{middle}, \text{ then } L_i = L_i^{old} \cap l_k, \ldots, l_s,\\
&L_i^{old} = L_i.
\end{aligned} \tag{23}$$

The idea of bisection of the set of alternatives has also been used in the CYCLE method [93] that is another classification method. The method builds "chains" of alternatives between two alternatives that are assigned to two different classes. The chains are constructed in such a way that two subsequent alternatives in the chain differ on one criterion value. Then, for each chain a subset of alternatives equidistant from the two limit alternatives of the chain is selected. Here, the bisection is applied in order to reduce the number of alternatives for which the informativeness is calculated, and in such a way to decrease the computational efforts during indirect classification by preference relation. Moreover, the alternative to be posed to the DM at each iteration is still selected based on the maximal informativeness rule (in the same way as in ORCLASS and SAC).

Similar idea of bisection is also used in several other methods for decoding of monotone function [93]. However, those methods comes from Boolean algebra logic and could be adapted to two classes problems only.

4.2.2 An Illustrative Example

In order to demonstrate the Dichotomic Classification method, we consider, as an example, a very small problem with two criteria $G = \{g_1, g_2\} = \{quality, price\}$ which importance is unknown to the DM, three values on each criterion $g_j = \{bad,\ satisfied,\ good\}$, and, for simplicity, only two classes $L = \{l_1,\ l_2\} = \{not\ \ to\ \ buy,\ to\ buy\}$ that are ordered in an increasing order. As we are solving a maximization problem, the best criteria values and the best class for the DM have maximal ranks. The size of the Cartesian product of the scales of criteria is equal to $|X| = 3^2 = 9$. The set of alternatives contains the following alternatives $X = \{(bad, bad), (bad, satisfied), (bad, good), (satisfied, bad), (satisfied, satisfied), (satisfied,\ good), (good, bad), (good, satisfied), (good, good)\}$. Thus, the task is to classify the nine alternatives into two classes. The Dichotomic Classification method performs the following operations.

The performance of the Dichotomic Classification method

Step 1 For each alternative $x_i \in X$ to be classified, assign the set of possible classes $L_i^{old} = L_i = L = \{l_1, l_2\}$. Allocate the alternative with all the best values, i.e., $x_9 = (good, good)$, into the best class $l_2 = to\ buy$, and the alternative with all the worst values, i.e., $x_1 = (bad, bad)$, into the worst class $l_1 = not\ to\ buy$.

Iteration 1

Step 2 At this point, it is checked whether there are not yet classified alternatives for which $|L_i| \neq 1$. There are seven not yet classified alternatives $X_L = \{x_2, \ldots, x_8\}$.

Step 3 The set of not yet classified alternatives is bigger than two, thus, we search for the middle alternative with regards to the best x_9 and the worst x_1 alternatives in the following way. We obtain the rank of the middle alternative on each criterion by averaging the ranks of the best and the worst alternatives:

$$r_{quality}(x_{middle}) = \frac{r_{quality}(x_{bad}) + r_{quality}(x_{good})}{2} = \frac{1+3}{2} = 2,$$

$$r_{price}(x_{middle}) = \frac{r_{price}(x_{bad}) + r_{price}(x_{good})}{2} = \frac{1+3}{2} = 2.$$

Thus, we should take as the middle alternative the alternative with the rank of two at the first criterion scale and with the rank of two at the second criterion scale. We obtain the first middle alternative $x_{middle} = x_5 = (satisfied, satisfied)$.

Step 4 The middle alternative is proposed to the DM for classification. Let us assume that the DM assigns the alternative x_5 to the class $l_2 = to\ buy$.

Step 5 There is no need for the inconsistency checking because we have only two classes (see Section 4.2.1).

Step 6 Next, the set of possible classes is recalculated for each not yet classified alternative according to the classification of the alternative at *Step 4*. We check the preference relation for the alternative classified by the DM at *Step 4* and each not yet classified alternative from the set X_L (and check whether there are not yet classified alternatives to which the alternative classified is preferred). We find that the alternatives $x_6 = (satisfied, good)$ and $x_8 = (good, satisfied)$ are preferred to the alternative $x_5 = (satisfied, satisfied)$ because $r_j(x_5) \leq r_j(x_i)$, $j = 1, 2$; $i = 6, 8$. According to the relation (12), the set of possible classes for the alternatives x_i, $i = 6, 8$ should be recalculated. For instance, we have $x_6 P x_5$, and, thus, $L_6 = L_6^{old} \bigcap l_2 = \{l_1, l_2\} \bigcap l_2 = l_2 = \textit{to buy}$. In such a way, these alternatives are classified indirectly: $x_6 \in l_2$, $x_8 \in l_2$. After this step, the process is repeated at next iteration.

Iteration 2

Step 2 We should check whether there are not yet classified alternatives. At this moment, the set of not yet classified alternatives is $X_L = \{x_2, x_3, x_4, x_7\}$.

Step 3 In this case, the set of not yet classified alternatives is bigger than two, and we again continue the search for the middle alternative on each of the two searching sets: the first set is defined with regards to the best and the middle alternatives obtained at the previous iteration, and the second set is defined between the middle and the worst alternatives in the same way as at the previous iteration. In our situation, we have the first middle alternative $x_3 = (bad, good)$ and from the second set only one alternative $x_7 = (good, bad)$ is left. We propose these alternatives to the DM for direct classification at the current and next iterations.

Step 4 The alternative $x_3 = (bad, good)$ is proposed to the DM for classification. Let us assume that the DM assigns it to the class $l_1 = \textit{not to buy}$: $x_3 \in l_1$.

Step 5 There is no need for the inconsistency checking because we have only two classes (see Section 4.2.1).

Step 6 Recalculation of the set of possible classes for not yet classified alternatives allows indirect classification of the alternative x_2 because it is less preferred than the alternative x_3.

After this step, we should go to the next two iterations *Iteration 3* and *Iteration 4* where at *Step 4* the DM will classify the alternative $x_7 = (good, bad)$, for instance, to the class $l_2 = \textit{to buy}$: $x_7 \in l_2$; and the alternative $x_4 = (satisfied, bad)$, for instance, to the class $l_1 = \textit{not to buy}$: $x_4 \in l_1$. Then, the classification is finished because there are no more not yet classified alternatives.

In such a way, the set of nine alternatives has been classified with four iterations (four questions posed to the DM) into two classes. Because we classify the whole Cartesian product of criteria scales, it is possible to define the boundary alternatives between classes in a similar way it is done for VDA methods (see Section 3.5.1). The boundary alternatives between the two classes are: $x_3 = (bad, good)$, $x_4 = (satisfied, bad)$, $x_5 = (satisfied, satisfied)$, and $x_7 = (good, bad)$. They allow the unambiguous classification of any alternative from the set X. For instance, if someone is interested in knowing to which class the alternative $x_6 = (satisfied, good)$ belongs, a comparison with the alternative $x_5 = (satisfied, satisfied)$ shows it belonging to the class l_2.

For the case where criteria are ordered according to its importance for the DM, for instance, in an increasing order, the classification procedure is simplified. The alternatives can be ordered from the very beginning of the classification procedure, and there is always only one boundary alternative between the classes (because the situations of incomparability and indifference between alternatives are excluded).

For our illustrative example, the alternatives in the set X are ordered from the very beginning in an increasing order and the set of corresponding ranks is available $R(X) = \{r_1, \ldots, r_9\} = \{1, \ldots, 9\}$. The middle alternative is selected. In our illustrative example at *Step 3* of *Iteration 1*, the rank of the first middle alternative is obtained as follows:

$$r_{middle} = \frac{r_1 + r_9}{2} = \frac{1+9}{2} = 5.$$

Thus, as a middle alternative we should take the alternative with the rank five. And then, the first middle alternative would be the alternative $x_5 = (satisfied, satisfied)$. *Step 5* and *Step 6* are modified in such a way that the relation of the strict preference between alternatives is substituted with the relation of strict preference between the alternative ranks. In our example, there is no need for checking inconsistencies because the number of classes is equal to two. Then, at *Step 6*, the recalculation of the set of possible classes for each alternative is done with regards to the relation (23). According to this condition, the alternatives x_i, $i = 6, 7, 8$ are classified indirectly to the class l_2: $x_6 \in l_2$, $x_7 \in l_2$, $x_8 \in l_2$. Then, the middle alternative is selected with respect to the alternatives x_1 and x_5. The rest of the procedure is repeated in a similar way.

4.2.3 Concluding Remarks and Future Research

The advantage of the Dichotomic Classification method for ordinal classification is in the absence of a need for any parametric information and/or assignment examples to be defined by the DM. This method is useful in cases where there is no information about assignment examples available. Basically, only the number of classes and their order as well as the criteria and their scales with numerical or verbal values should be known in advance.

When compared to the VDA methods developed earlier, such as SAC, Dichotomic Classification is computationally less complex. Improvement of the effectiveness and computational complexity allows overcoming the limits of the SAC method for the size of the set of alternatives to be classified and the number of the criteria on which these alternatives are evaluated. The questions posed to the DM for the direct classification may not be the most difficult ones as in the ORCLASS and SAC. We have also tried to avoid embedded heuristics (about uniform distribution of the centers of unknown classes) of the other VDA methods and create a method that works efficiently for different distributions of alternatives between classes.

We also considered the possibility of establishing a lexicographic order on the set of criteria. This information, if available, simplifies the classification procedure and reduces the number of questions posed to the DM for direct classification.

In the future, we would wish to study the possibility to use criteria weights in VDA or the possibility for the DM to select a subsets of the most important criteria to be used for classification. Another possible extension is the application of assignment examples in VDA; naturally, this should simplify the classification process. Moreover, as we have pointed out in Section 3.5.1, in existing VDA methods, the actual support to the DM when inconsistencies appear is absent. In future work, we wish to study patterns of human behavior and possibilities to provide real support to the DM for guiding him or her towards resolving inconsistencies. For instance, it may be possible to track the previous DM's answers when performing classification and search for some behavioral strategy (such as considering several criteria as the most important ones for classification to some class). Then, in case of inconsistencies, the DM can be provided with guidelines for classifications which support or oppose the strategy used in previous steps. In addition, some experience from studying human behavior when solving choice type of MCDA problems [51] can be adapted here.

5 COMPARISON OF METHODS AND SELECTION OF A METHOD

As can be seen from the previous chapter, a large number of methods have been developed for classification in MCDA literature. Except for the reviews of MCDA methods for classification made in like [26], [37], [161], [172], there are not many surveys of MCDA approaches to classification available in the literature. Even less attention is paid to the comparison of different methods. Comparing of different methods also allows us to provide some guidelines for the selection of a method and recommendations for the method to be used in a particular situation. In this chapter, we try to fill the gap to some extent and compare pairs of the methods as well as provide recommendations for the selection of a method.

In this work, we have restricted our comparisons to the theoretical aspects of different methods. However, it should be mentioned that in addition to the theoretical comparison of similarities and differences of the methods, it is important to compare them when applied to the same real-life problems. In [37], some MCDA methods for classification presented here are compared in an experimental way when applied to the same artificial data sets as well as to real-life problems.

5.1 Comparison of Methods

In the introduction to the previous chapter, we categorized the discussed MCDA methods for classification into three main groups: the first type is direct parameter-based methods (there can be utility function-based and outranking relation-based direct methods); the second type is indirect parameter-based methods; and the third type is parameter-free decision rule methods. In this section, we compare some pairs of methods in each category.

When comparing the features of the surveyed methods, we have tried to define their advantageous and disadvantageous properties as well as similarities and peculiarities. The comparison of existing methods also motivates us to improve them and to create some new methods that are presented in Chapter 4.

5.1.1 Comparison of Utility Theory and Outranking Approach

In the direct parameter-based methods approach for the construction of a model, the alternatives for an MCDA type of problem (see Section 2.1.6) are estimated in different ways. Indeed, the utility theory suggests to calculate an absolute performance of each alternative on all criteria and then compares such overall utilities of all the alternatives for solving one or another MCDA problem type. The outranking approach (see Section 3.1), on the other hand, proposes to estimate the relative performance of pairs of alternatives, in order to discover the preference relations on such pairs or find out that alternatives are indifferent or incomparable. Both of these methodological approaches have different assumptions and requirements to the initial and resulting information as well as to the process of decision aiding itself.

Both the utility theory and the outranking approach are direct methodologies that require the DM to provide parameter values in the form specified by the model. However, these methodologies construct the decision aiding model in different ways and are based on different assumptions. Even though both of them may be used for solving any of the MCDA problem types (see Section 2.1.6), usually they are applied in different situations depending on the initial information available and requirements to the form of results and the decision aiding process itself. In utility theory, each alternative is estimated independently of the other alternatives and an overall utility is calculated as an aggregation of its performance on each criterion. The solution of a problem is done through a comparison of absolute utilities of alternatives. On the other hand, in outranking methods, a relative pairwise comparison of alternatives is done. The solution of a problem is based on the values of outranking indices between the compared alternatives. The outranking concept considers one alternative "to be as good as" another one if there are enough criteria with regards to which the first alternative is better than the second one; and at the same time there is no "big enough" opposition to such preference, that is, there are no criterion that interposes a veto against such a preference. The concept "big enough" is defined with a cutting level.

The assertion "to be as good as" is defined with preference and indifference relations. When compared to utility theory, in outranking methods, there is no need for a strict relation of preference or indifference between the alternatives to be held: they may be in a relation of weak preference. Two alternatives are assumed to be in a relation of weak preference, when the DM hesitates to define the preference of one alternative over another one or to consider them indifferent. On the other hand, two alternatives may be incomparable, if for some reason (for instance, in case of very conflicting alternatives) the DM cannot define a relation of preference (strong or weak) or indifference between them. By introducing the incomparability of two alternatives, the situation that one alternative is performing very well on some criteria and very poorly on other ones is taken into account. This means that there strongly conflicting criteria. In addition, some uncertainty in the output data may remain. Such approach has an advantage if the DM has difficulties for resolving uncertainties; however, it may be not applicable should the output of the model have no uncertainties and should it provide the DM with exact result.

When compared to the utility theory, the outranking approach does not assume the preference and indifference relations to be transitive. On the other hand, in outranking methods, the criteria may be verbally defined or even have categorical scales of values, when compared to the utility theory where all non-numerical estimations should be transformed into a single scale $[0, 1]$. However, when compared to utility theory, in outranking methods, more preference information is required from the DM. Thus, not only the alternatives estimated on the criteria values and criteria weights, but also preference, indifference and veto thresholds as well as cutting level are elicited from the DM.

One more basic distinction between the utility theory and the outranking approach is interpretation of weights. In the utility theory that is considered to be a compensatory approach, the weight of the criterion represents a trade-off, such that an improvement of this criterion by one unit on the scale of criterion values would lead to a deterioration on the scale of other criterion or criteria with the number of units indicated by trade-off. In contrast, the outranking approach is a non-compensatory approach that means that it does not consider trade-offs between criteria. In the outranking approach, the weight indicates votes that support or oppose the assertion "one alternative is at least as good as another alternative".

The main disadvantage of the outranking approach is the cognitive loading of the DM when inferring all the parameters of the model, even though this process may help the DM to understand the model better and explore his or her own preferences. This is due to the fact that the DM rarely can give precise numerical values for the parameters of the model. Another difficulty appears when aggregating all the preferences according to the concordance and non-discordance principle. The details of this process are not intuitive enough for the DM, and for instance, may lead to non-monotonic results of ranking. Moreover, in ranking or choice type of MCDA problems, it could be difficult for the DM to understand the changes in the final decision that appear with the addition or removal of some alternatives to the set of given alternatives. This problem, however, does not appear in classification type of MCDA problems.

5.1.2 Comparison of PDA and SMAA

In view of the fact that both of the direct parameter-based methodologies, utility theory and outranking approach, require cognitive loading of the DM when eliciting parameters of the models, two other indirect parameter-based methodologies appear: PDA and SMAA. Both of the indirect parameter-based methodologies aim at assisting and simplifying preference elicitation in terms of parameters of an MCDA model.

A simulation approach is used for modeling different preferences in SMAA (see Section 3.4). These methods are applied in the situations of uncertainty and imprecision of the initial information obtained from the DM; the case of a complete absence of preferential information is also taken into account. Different preferences are modeled for each of the possible decisions, and the most probable values are proposed to the DM for evaluation. Thus, situations analyzed with SMAA are similar to ones observed in PDA. However, in SMAA, we do not have to have assignment examples. On the other hand, if some partial information about preferences (in terms of parameters of the model or assignment examples) is available at the beginning of the decision aiding process, it may be used in the SMAA methods.

5.1.3 Comparison of PDA and VDA

When compared to the indirect parameter-based PDA methods, the VDA methods are parameter-free and so-called decision rule methods. The use of both the utility function and the outranking model is possible when developing the model of preferences with PDA. On contrast, in VDA, there is no any specific formal model used, but the set of classification rules is discovered.

The PDA assumes that the DM has some earlier made decisions, so-called assignment examples, for instance, from his or her past experience or has developed them in some other way (see Section 3.3). Then, a model is constructed using mathematical programming techniques in such a way that earlier decisions provided by the DM are resolved as close as possible (with minimal error) by this model. Such an approach assumes that the earlier decisions provided by the DM are considered to be good representatives of the problem situation.

On the other hand, the VDA methods (see Section 3.5.1) assume that it is difficult initially to have a set of assignment examples that represents the decision situations completely enough or such assignment examples may be absent in the situation which is new for the DM. The methods also assume a particular preferential system for the DM. However, they suppose that the preferences of the DM are not stable and that they may evolve over time. That is why the methods are interactive and here the DM is involved in the process of model construction. He or she provides some part of his or her preferences at each step of the interactive procedure. The information provided is used for the development of the model until it satisfies the DM. Preferences are allowed to change during the elicitation process, and that is why contradictions may appear. They should be resolved within the VDA method.

An advantage of using an additive utility function in the framework of PDA is the possibility to treat verbal estimations without transforming them into a numerical analogy in the same way as in the interactive VDA methods (see Section 3.5.1). One more similarity of the PDA and VDA methodologies is their indirect nature: they work with the earlier decisions provided by the DM. However, in PDA the set of assignment examples is provided before the actual work of the method starts. On the other hand, in VDA, at each step of the interactive dialog the DM has to make some decisions. For the classification problem, the DM has to assign one alternative selected by the method at each iteration. The selection of the next alternative to be proposed to the DM for classification depends on the earlier made decisions. Thus, it would be interesting to see whether a set of boundary alternatives obtained at the end of the VDA interactive procedure would produce the same results of classification when compared to the results of a PDA based method. It is also interesting to see if two models based on the utility function and the outranking relation, for instance UTADIS and ELECTRE TRI Assistant, generate the same classification results when the same set of assignment examples alternatives is applied in PDA.

5.1.4 Comparison of VDA and Outranking Approach

When compared to the direct parameter-based outranking approach methods, the VDA methods are parameter-free decision rule methods. In the same way as in the outranking approach, in VDA, the pairs of alternatives are compared and preference and indifference relations are detected. However, in VDA, all the relations are transitive, and the incomparabilities are considered due to the human fallibility, but these should be resolved with the DM. On contrast, in the outranking approach, intransitivity and incomparability of preferences as well as weak preference are allowed. Both approaches allow verbal values to be used, but, in the outranking approach, the verbal definition of the preference, indifference and veto thresholds may be difficult for the DM; not to forget the weight information that usually has numerical format and a cutting level that is the technical parameter and has to be defined in the interval $[0, 1]$.

5.1.5 Comparison of VDA and Utility Theory

There are many similarities between VDA and the utility theory. Indeed, they both assume a possibility to compare all alternatives based on indifference and preference relations and allow their transitivity. All incomparabilities have to be resolved in both methodologies. The comparison of these two approaches [112] has been based on an application to the same problems (see [49], [92], [94]). In VDA as well as in the utility theory, verbal values on criteria scales are allowed. However, in VDA they are not reduced to numerical output as it is done in the utility theory.

5.1.6 Comparison of Fuzzy and Rough Sets

Both fuzzy and rough sets work in a situation of uncertainty. However, while RSA model imprecision in the definition of classes, fuzzy sets assume the belonging of an alternative to a class to be unsure. On the other hand, both methods apply the concept of membership function, and even though it has the same meaning (the degree to which each alternative belongs to a class), in the fuzzy sets theory the shape of the membership function is subjectively assumed, while in RSA it is calculated from the data available. In both approaches, fuzzy sets and

RSA, partial information can be enough for classification of alternative. For instance, in RSA, a subset of possible attributes and/or criteria may be considered sufficient for making a decision about classification.

5.1.7 Comparison of VDA and Rough Sets

There are a lot of similarities between VDA and RSA. Both parameter-free methodologies solve the classification problem by construction of a set of decision rules, and both of them can be applied for nominal as well as ordinal types of classification problems. (Even though nominal classification could be handled, the methods are less efficient in this case.) They both work in a natural language and operate with qualitative values. There is a possibility to check inconsistencies in both methods as well as a tool for providing explanations for classification results.

On the other hand, a number of differences exist. RSA assumes that there are assignment examples from the past experience of the DM or decisions made for realistic alternatives not considered in the set of alternatives given by the DM, while the VDA methods are interactive and ask the DM to classify alternatives selected by the VDA method. Usually, the VDA method selects the most difficult alternatives for classification, in contrast to RSA, where the DM has to provide assignment examples that are usually the most evident cases. In VDA, a set of rules defines the boundary alternatives between classes, while in RSA, there are, usually, many more rules that cover the exact or approximate classification of assignment examples estimated on the whole set or subsets of criteria and/or attributes. Most of the VDA classification methods were developed for ordinal classification, while RSA has been initially developed for nominal type of classification and has been later adapted for ordinal type of classification. In VDA, the alternatives are estimated on a whole set of criteria, while in RSA the alternatives may be evaluated with a whole or subset of criteria and/or attributes. In VDA, after the boundaries are detected, we can assign any alternative unambiguously, and the classification is crisp. While in RSA the membership function defines one of several possible classes and the final solution about classification may still contain some ambiguity.

5.1.8 Comparison of MC Filtering, PROAFTN and TRINOMFC

The MC Filtering and PROAFTN methods are similar in a sense that using the interval $[g_j^{min}(b_q^h), g_j^{max}(b_q^h)]$ with $g_j^{min}(b_q^h) \leq g_j^{max}(b_q^h)$ in PROAFTN is equivalent to using an average value on each criterion $g_j(b_q^h) = (g_j^{max}(b_q^h) + g_j^{min}(b_q^h))/2$ and indifference threshold $q_j(b_q^h) = g_j^{max}(b_q^h) - g_j(b_q^h) = g_j(b_q^h) - g_j^{min}(b_q^h)$ in MC Filtering. The preference $p_j(b_q^h)$ and discrimination d_j^+ and d_j^- thresholds are very similar: both types of thresholds show the smallest difference between two alternatives such that one alternative is considered to be preferred to the other one on the criterion g_j.

Even though there is no indifference threshold $q_j(\cdot)$ (required in MC Filtering) used in PROAFTN, the interval of given values on each criterion plays the same role and shows all the possible criterion values that are considered equally possible for the alternative. On the other hand, using intervals is very convenient in cases where giving precise values may not be possible. This feature allows a flexible definition of the set of reference alternatives for classes in the PROAFTN method when compared to MC Filtering where crisp values on criteria are required. There is a need for the definition of preference $p_j(\cdot)$ and veto $v_j(\cdot)$ thresholds in both MC Filtering and PROAFTN.

When compared to MC Filtering and PROAFTN that are based on concordance and non-discordance principles, in TRINOMFC, the idea similar to the classical approach (e.g., statistics and artificial intelligence) to nominal classification is used. Indeed, for the nominal type of classification, it is more important to estimate the similarity and dissimilarity as it is done in TRINOMFC than preference between two alternatives used in MC Filtering and PROAFTN. In TRINOMFC, there is no need for a veto threshold to be specified. On the other hand, instead of the preference and indifference thresholds used in MC Filtering (or similar idea with fuzzy criteria values of reference alternatives in PROAFTN), in TRINOMFC, the similarity and dissimilarity thresholds are used. Similarity threshold has the same meaning as indifference threshold in the outranking theory (on which MC Filtering and PROAFTN are based). However, the dissimilarity threshold shows a minimal value of difference between two values on a criterion that makes one alternative to be dissimilar to another one (when compared to the preference threshold there is no preference of one alternative over another one). Thus, such thresholds are more natural for nominal classification problems.

In MC Filtering, the exact meaning of weights is not specified, but, since the outranking relation is used throughout, we may suppose that it is the same as in the outranking methods at least for ordinal classification. Then, the weight of the criterion indicates the relative importance of the criterion when compared to other criteria in terms of votes that support or oppose the assertion "one alternative is at least as good as another alternative". However, from the examples provided in [131] for nominal classification, there were different weights for each class. From this, we can guess that the authors assume that the weight for nominal classification indicates the relative importance of the criterion when compared to the other criteria that support the assignment in the class. In PROAFTN and in TRINOMFC, however, the meaning of weights corresponds more to the classification problem itself than to the comparison of pairs of alternatives. Indeed, in these methods, the weight indicates the relative importance of a criterion when compared to other criteria in terms of votes that support or oppose the assignment in a particular class.

Another important issue is treating attributes in nominal classification. All three methods, MC Filtering, PROAFTN and TRINOMFC, work only with criteria, even though it is natural for nominal classification to have some attributes (that are neither minimized nor maximized) to be taken into account when estimating alternatives to be classified. Even though in MC Filtering the attributes are mentioned, it is not very clear how they are treated. On the other hand, even though in TRINOMFC the attributes are not mentioned, it is very easy to adapt the method to use them with the true-type similarity function.

5.1.9 Representation of Weights in Different MCDA Theories

An important issue of representation of weights in different MCDA models is discussed in different chapters of [16]. The authors emphasize that intuitive weights are not appropriate in all MCDA models. In the linear utility theory, the weights represent trade-offs between pairs of criteria. For instance, for any two criteria g_j and g_z the ratio of their weights w_j/w_z shows the increment in criteria values on the scale of the criterion g_z that compensates the loss in the criteria values on the scale of the criterion g_j. Such a ratio depends on the scaling of both the criteria and should be adapted if any scaling changes occur. In the outranking approach, the weights have a different meaning. Indeed, the weights represent the "voting power" that supports each criterion rather than trade-offs between criteria. Moreover, the veto threshold that is used in the outranking approach does not allow any trade-offs between criteria.

Even though the interpretation of weights varies from method to method, it still may be different from the intuitive meaning of weights for the concrete DM. People often express their preferences independently from the context of the current problem. Thus, when the context changes, the weights do not change in their opinion, even though they should according to the situation.

In addition, the intuitive modeling of weights may be different even for the same DM relative to the reference point used for the representation of a criterion. In [76], the authors show that humans react differently to the stimuli presented in terms of losses when compared to the same stimuli defined in terms of gains. If a criterion is framed in terms of losses depending on some reference point, it most probably will have a higher weight when compared to the same criterion defined in terms of gains relative to another reference point. On the other hand, splitting or hierarchical structuring of the criteria may lead to under/overevaluation of weights [160]. Thus, there is a danger of mistreating weights from the view point of model peculiarities as well as of their misinterpretation influenced by the behavioral aspects of humans.

It should also be mentioned that, in the SMAA methods, the weights interpretation depends on the model used. Thus, for the utility based methods, like SMAA and SMAA-2, the weights represent trade-offs, while for the outranking based methods, like SMAA-3 or SMAA-TRI, the weights define the "voting" power of each criterion. On the other hand, in the SMAA-Classification method, a weight vector shows the importance of each criterion in the distance function for classification of the alternative into the class. Indeed, if it is important that the criterion value of an alternative is classified close to the corresponding reference alternative value, then the corresponding weight should be big.

Another question is: should the meaning of the weights be different for the nominal and ordinal classification? As we can see from the methods developed so far, it is different. Indeed, the ordinal classification methods assume a set of weight to be defined for the whole set of attributes and/or criteria; however, for nominal classification the weights are different for each class.

5.2 Selection of Classification Method

There is no single method that suits best to any classification problem. That is the reason for providing here some recommendations for selecting a method to be used. The selection of the method depends on the available initial information, on the requirements to the results of classification and the decision aiding process itself, not forgetting the difference between ordinal and nominal types of problems. The ability of the DM to define parameters of the model and his or her opinion about the method should also be considered. The method that best suits the DM's preference structure and his or her way of thinking should be selected.

Here, we present some guidelines for selecting a method among the surveyed ones. The recommendations provided are very subjective and were created by the author for the following reasons: on the one hand, there is a gap in the MCDA literature when searching for guidelines as to which method to use for a particular situation, and, on the other hand, there is an evident need of such guidelines, for instance, if the analyst wish to follow the constructive approach to the decision aiding (see Section 2.2.2). These recommendations do not intent to be objective but should be seen as the first attempt for development in this direction.

As has been mentioned earlier, the selection of the MCDA method is an MCDA problem itself. That is why we have to define the criteria (that are the main properties of the methods) that should be taken into account when making the choice of a method. The idea is to consider assumptions made in each method as well as the ability of the DM to provide information needed for each method.

As has already been mentioned in the introduction to Chapter 3, the first classification methods in MCDA stem from the ranking problem and have been developed for ordinal classification problems. The methods for ordinal and nominal types of classification are different in the sense that for ordinal classification, the additional information about the order between classes can be used and it may improve and/or simplify the procedure of classification. Moreover, in ordinal type of classification problems, the preferences between alternatives are evaluated; this contrasts with nominal classification problems, where indifference or similarity between alternatives is estimated. That is why we assume that the first step when selecting the MCDA method for classification should be defining the type of classification (ordinal or nominal). However, here the question appears: can we apply the MCDA methods that are developed for solving nominal classification problems for assigning alternatives to classes that are ordered (but for some reasons we are not aware about the information about the order)? We will get back to this question in connection with numerical experiments in Section 6.1.

We have presented two decision trees for the selection of the MCDA method for ordinal and nominal types of classification in Figures 16 and 17, respectively. Most of the questions to be asked when selecting a method for ordinal and nominal types of classification are the same. However, there are different methods developed for the same purposes for ordinal and nominal types of classification. The decision trees should be self-explaining. In the nodes, there are questions that one may encounter when selecting a method to be used for classification with MCDA. When developing these decision trees, the idea was to use only the answers "yes" or "no", and assume the answer "no", in the case the answer was "I do not know".

As can be seen from the trees, when selecting a method we consider the categorization of the MCDA methods provided in Chapter 3 (see Figure 4) that takes into account the ways to describe the behavior of the DM by means of some formal or judgemental model and the way the preferences of the DM are elicited. In this respect, the opinion of the DM is very important. The analyst should select a method that is suitable for the DM and that corresponds to his or her knowledge and system of preferences. We have to check what kind of model resembles the DM's way of thinking and whether the DM can easily define the parameters of such model or another one. If some model corresponds to the DM's way of thinking, one of the direct methods can be used (that may be utility function-based or outranking relation-based). However, if there are difficulties with the definition of parameters, we have to consider indirect parameter-based methods. Otherwise, if there is no way to specify any suitable formal model, parameter-free methods should be used.

Trees include the methods, discussed in more detail in Chapter 3, as squares with a solid line and some less considered methods as squares with dashed line. We have also put the mark $\times$ in the squares for which we did not find a method that would correspond to the considered properties.

In addition to the assumptions made in each method and the abilities of the DM to provide information needed for each method discussed in the decision trees, it is important to consider how methods treat basic concepts (alternatives, criteria, classes, etc.) as well as the way decision aiding is accomplished in the different MCDA methods for classification. Such analysis of good and weak properties has been done in the concluding remarks to the methods presented in Chapter 3. We summarize the main properties in Table 4.

Here, the mark $\times$ indicates the presence of a feature in the method; the mark $\times^*$ defines that this property is not necessary present; the $\times^f$ shows that for these properties the values are fuzzy or $\times^{f*}$ may be fuzzy; $\times^m$ denotes a membership function that is not a threshold but, in some sense, similar technical information. The properties are classified into positive ("+"), negative ("-") or neutral ones.

Mainly, the following aspects of the models have been considered: type of measurements on the scales of criteria (verbal or numerical values; and the latter ones, in turn, are presented as exact values, or intervals, or distributions of values) defining classes by boundary/reference alternatives, or by assignment examples,

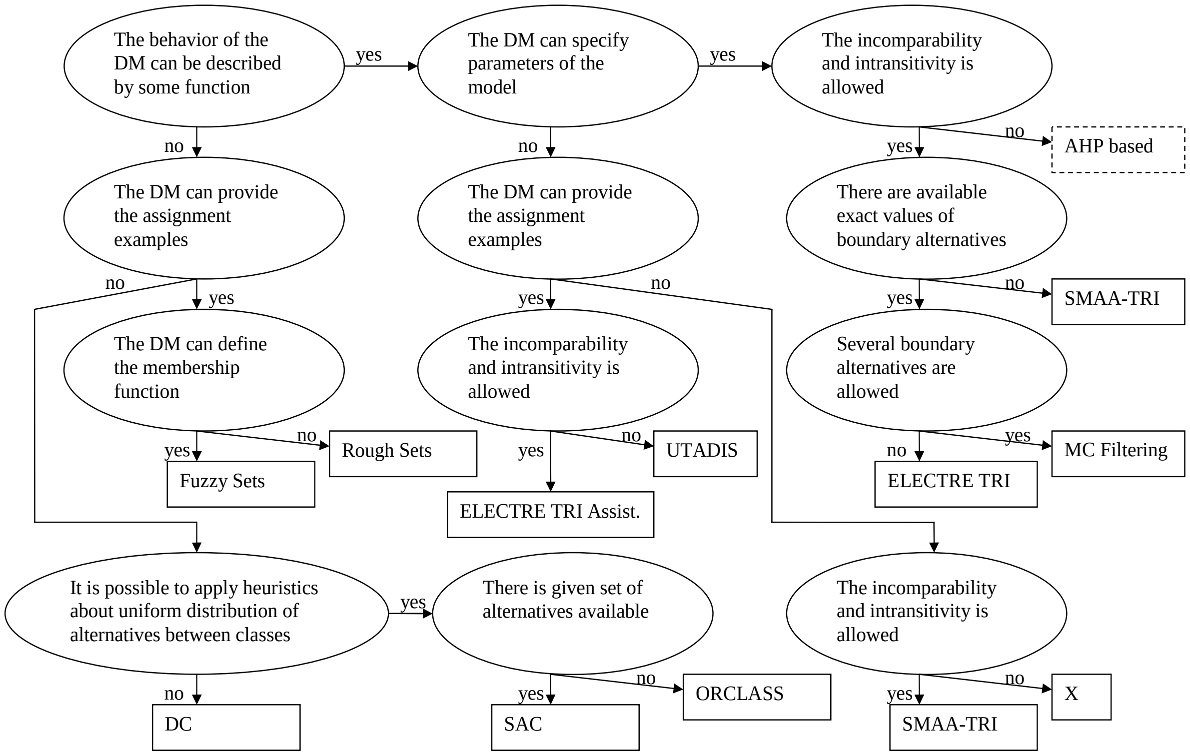

FIGURE 16 Selecting an MCDA method for ordinal classification

FIGURE 17 Selecting an MCDA method for nominal classification

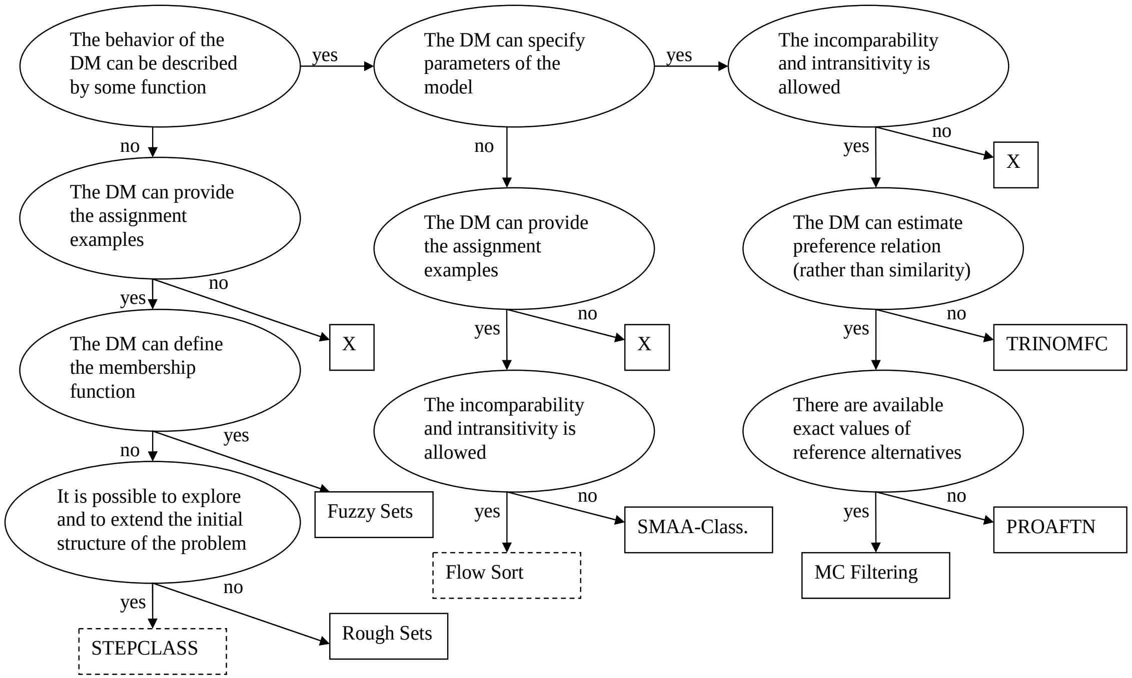

TABLE 4 Properties of MCDA methods for classification

	ELECTRE TRI	ELECTRE TRI Assist.	UTADIS	SMAA-TRI	ORCLASS	SAC	DC	Fuzzy Sets	Rough Sets	MC Filtering	PROAFTN	TRINOMFC	SMAA-Classification
- Thresholds are needed	×			$\times^{f*}$				$\times^{m}$		×	×	×	
- Weights are needed	×			$\times^{f*}$						×	×	×	$\times^{*}$
- Boundary or reference alternatives are needed	×			$\times^{f*}$						×	$\times^{f}$	×	
- Assignment examples are needed		×	×					×	×				×
+ Attributes may be considered								×	×				×
- Difficult assignments are asked					×	×	$\times^{*}$						
+ Inconsistencies are checked					×	×	×		×				
+ Verbal values are not adjusted to numbers in output					×	×	×	×	×				
+ Intervals of values are allowed				×							×		×
+ Distributions of values are allowed				×									×
+ Modeling of absent preferences				×									×
+ Interactivity				$\times^{*}$	×	×	×						$\times^{*}$
+ Large data sets				×			×						×
+ Classification based on the subsets of criteria or attributes									$\times^{*}$				$\times^{*}$
+ Possibility to check dependencies between criteria or attributes									×				
+ Negative examples may be considered									×				
Assignment to exactly one class			×		×	×	×						

or not predefined (but their number is assumed to be known); assignment to exactly one class or probability of assignment to each class is enough; and others.

We use some general preassumptions, such as "it is better if as little information is asked from the DM as possible"; and "the more possibilities to consider different types of information (e.g., ways to measure criteria or to consider fuzzy values) the better the method is".

6 NUMERICAL EXPERIMENTS

In this chapter, we present some numerical experiments carried out with the two new methods, SMAA-Classification and Dichotomic Classification, introduced in Sections 4.1 and 4.2, respectively. We perform tests on the data sets available in the literature and compare our results to the results obtained with methods developed earlier. It should be mentioned that with few exceptions, like [37], there are not many results of solving classification problems with MCDA approaches documented in the literature. That is why we experienced difficulties when searching for the data sets for our numerical experiments, especially, for testing and comparing the SMAA-Classification method for nominal classification problems. Next, we describe our numerical experiments in more detail.

6.1 SMAA-Classification

In Section 4.1, we have introduced the SMAA-Classification method for nominal classification of a set of given alternatives to a predefined set of classes. It is assumed that the set of given alternatives estimated on a set of criteria is available, and that for each class at least one assignment example exists. SMAA-Classification uses the Monte-Carlo method for simulation of uncertain or unknown preferences. At each iteration, taking into account the simulated preferences, the distances between each alternative to be classified and each assignment example of each class are calculated. The alternative is assigned to the class for which the distance is minimal. After a specified number of simulation runs, statistical information is collected about acceptabilities of each alternative to each class and about typical preferences that support each assignment.

Next, we describe some numerical experiments that we have done in order to test the SMAA-Classification method and compare it to the results obtained with the other methods. From the survey presented in Chapter 3, it can be seen that there are few methods developed for nominal classification in the framework of MCDA. The reasons for this, that have already been mentioned, are due to the orientation of the MCDA methods towards solving problems of preferential nature. Even less is documented about tests of methods on data sets or applications to real-life problems. That is why we have experienced difficulties in finding some testing data.

However, the nominal classification problem can be considered to be more general when compared to the ordinal type, thus, we can compare the results of a more specified case to a more general one, but not vise versa. That is why, in this work, apart from comparing the results of two nominal classification methods: SMAA-Classification and TRINOMFC, on the same data set, we also compare SMAA-Classification for nominal classification and ELECTRE TRI for ordinal classification on other data sets. In a similar way, in [37], the ELECTRE TRI and UTADIS methods for ordinal classification have been compared to the statistical discriminant analysis methods developed for nominal classification.

6.1.1 Analysis of Results

In order to test the performance of the SMAA-Classification method, we have done several numerical experiments with data sets widely-used in MCDA publications. We compare the results of SMAA-Classification to the results obtained with ELECTRE TRI and TRINOMFC. Even though ELECTRE TRI is an ordinal classification method, it is possible to compare the results obtained with it to the results obtained with nominal classification methods such as SMAA-Classification. In this case, we have to assume that in the data provided the order of classes is unknown and assignment examples for each class are available. On the other hand, in this work, we have used testing data sets with precise criteria values of the alternatives to be classified, while the SMAA-Classification method can also perform assignment on data sets with imprecise or uncertain criteria values.

The following applications are considered: diagnosis of firms' health [34] which has been used for the estimation of the ELECTRE TRI method in [115]; credit granting in the banking sector introduced in [165] and used in [120] for testing ELECTRE TRI; and identification of accident type [101] used in the evaluation of the TRINOMFC method. We ran the Monte-Carlo simulation in the SMAA-Classification method with 10000 iterations for each application.

The results of testing three applications with SMAA-Classification, ELECTRE TRI and TRINOMFC, are presented. Tables 5, 10, and 14 introduce the assignment examples for each problem. In these tables, the first columns define the alternatives estimated on the sets of criteria and the second columns show the assignment class. The alternatives to be classified and the results of classification by different methods are presented in Tables 6-9, 11-13, and 15. In these tables, the first columns show the alternatives to be classified estimated on the set of criteria and the rest of the columns show the classes where the alternatives have been assigned by different methods. For the SMAA-Classification method, we have presented the acceptability indices for the alternative to be assigned into each class. The cells highlighted in bold face indicate the highest acceptability indices when compared to the other classes.

TABLE 5 "Diagnosis of the health of the firms": assignment examples

Reference Alternative	Class
b_1^1 =(-12.0, -62.5, 92.5, 29.5, 42.5, 0.5, 0.0)	l_1
b_2^1 =(-5.0, -50.0, 82.5, 25.5, 36.0, 1.5, 1.0)	l_2
b_3^1 =(4.0, -10.0, 67.5, 20.5, 27.0, 3.0, 2.5)	l_3
b_4^1 =(16.5, 25.0, 47.5, 14.0, 18.0, 4.5, 3.5)	l_4
b_5^1 =(30.5, 48.5, 27.0, 5.0, 7.0, 5.0, 4.5)	l_5

The first application "Diagnosis of the health of the firms" involves 40 firms evaluated on 7 criteria and assigned into 3 ordered classes defined with a set of 3 assignment examples (one for each class) [34]. We have compared the performance of SMAA-Classification to the results of the classification given by ELECTRE TRI and presented in [115]. In Tables 6-9, the first column indicates the alternative to be assigned, the second and the third columns show the classes for the indicated alternative according to the assignment with the ELECTRE TRI optimistic and pessimistic procedures, respectively. The last columns show the classes and acceptability indices for each alternative to be assigned into each class obtained with the SMAA-Classification method.

The results of the SMAA-Classification method are compatible with the results obtained by the ELECTRE TRI method. For both of these methods, the class l_1 is empty. The rest of the classifications differ in the following ways. In cases where the assignment according to the optimistic and pessimistic procedures of ELECTRE TRI is different, SMAA-Classification usually shows the highest acceptability into one of them (for instance, the alternative x_1 is assigned into class l_4 by the ELECTRE TRI optimistic procedure and also by SMAA-Classification, and into l_5 by the ELECTRE TRI pessimistic procedure). In this respect, the general picture of classification with SMAA-Classification and ELECTRE TRI is the same.

The second application "Credit granting in the banking sector" involves 45 alternatives evaluated on 7 criteria and assigned into 3 ordered classes with a set of 13 assignment examples (several for each class) [165]. We have compared the work of SMAA-Classification to the ELECTRE TRI results presented in [120]. The results of SMAA-Classification are similar to the ones of ELECTRE TRI (see Tables 11-13).

The third application "Identification of accident type" is very small: there are 3 workers evaluated on 3 criteria and assigned into 3 unordered classes with a set of 6 assignment examples (several for each class). We have compared the results of TRINOMFC presented in [101] with the results obtained by SMAA-Classification. As we can see in Table 15, the first two alternatives are assigned similarly by both methods, while the third one is assigned into the class l_3 by SMAA-Classification and into the class l_2 by TRINOMFC (although the similarity index for the class l_3 is also high).

TABLE 6 "Diagnosis of the health of the firms": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE		SMAA-Classification	
	PES	OPT	Class	Acceptability index
x_0 =(35.8, 67.0, 19.7, 0.0, 0.0, 5.0, 4.0)	l_5	l_5	l_5	a_0^5**=1**
x_1=(16.4, 14.5, 59.8, 7.5, 5.2, 5.0, 3.0)	l_4	l_5	l_4	a_1^4**=0.916**
			l_5	a_1^5=0.084
			l_3	a_1^3=0.00001
x_2=(35.8, 24.0, 64.9, 2.1, 4.5, 5.0, 4.0)	l_3	l_5	l_5	a_2^5**=0.698**
			l_4	a_2^4=0.301
			l_3	a_2^3=0.001
x_3=(20.6, 61.7, 75.7, 3.6, 8.0, 5.0, 3.0)	l_3	l_5	l_4	a_3^4**=0.668**
			l_5	a_3^5=0.312
			l_3	a_3^3=0.02
x_4=(11.5, 17.1, 57.1, 4.2, 3.7, 5.0, 2.0)	l_3	l_5	l_4	a_4^4**=0.844**
			l_5	a_4^5=0.124
			l_3	a_4^3=0.031
x_5=(22.4, 25.1, 49.8, 5.0, 7.9, 5.0, 3.0)	l_4	l_5	l_4	a_5^4**=0.712**
			l_5	a_5^5=0.288
x_6=(23.9, 34.5, 48.9, 2.5, 8.0, 5.0, 3.0)	l_4	l_5	l_5	a_6^5**=0.513**
			l_4	a_6^4=0.487
x_7=(29.9, 44.0, 57.8, 1.7, 2.5, 5.0, 4.0)	l_4	l_5	l_5	a_7^5**=0.811**
			l_4	a_7^4=0.189
x_8=(8.7, 5.4, 27.4, 4.5, 4.5, 5.0, 2.0)	l_3	l_5	l_4	a_8^4**=0.708**
			l_5	a_8^5=0.271
			l_3	a_8^3=0.021
x_9=(25.7, 29.7, 46.8, 4.6, 3.7, 4.0, 2.0)	l_3	l_5	l_4	a_9^4**=0.656**
			l_5	a_9^5=0.334
			l_3	a_9^3=0.01
x_{10}=(21.2, 24.6, 64.8, 3.6, 8.0, 4.0, 2.0)	l_3	l_5	l_4	a_{10}^4**=0.829**
			l_5	a_{10}^5=0.111
			l_3	a_{10}^3=0.06
x_{11}=(21.2, 24.6, 64.8, 3.6, 8.0, 4.0, 2.0)	l_3	l_5	l_4	a_{11}^4**=0.756**
			l_5	a_{11}^5=0.226
			l_3	a_{11}^3=0.019
x_{12}=(21.2, 24.6, 64.8, 3.6, 8.0, 4.0, 2.0)	l_3	l_5	l_5	a_{12}^5**=0.588**
			l_4	a_{12}^4=0.404
			l_3	a_{12}^3=0.008
x_{13}=(21.2, 24.6, 64.8, 3.6, 8.0, 4.0, 2.0)	l_3	l_5	l_4	a_{13}^4**=0.852**
			l_3	a_{13}^3=0.132
			l_5	a_{13}^5=0.017

TABLE 7 Cont. "Diagnosis of the health of the firms": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE		SMAA-Classification	
	PES	OPT	Class	Acceptability index
x_{14}=(10.4, 9.3, 80.9, 1.4, 4.1, 4.0, 2.0)	l_2	l_5	l_4	a_{14}^4**=0.598**
			l_3	a_{14}^3=0.317
			l_5	a_{14}^5=0.085
			l_2	a_{14}^2=0.00001
x_{15}=(17.7, 19.8, 52.8, 7.9, 6.1, 4.0, 4.0)	l_4	l_5	l_4	a_{15}^4**=0.88**
			l_5	a_{15}^5=0.12
x_{16}=(14.8, 15.9, 27.9, 5.4, 1.8, 4.0, 2.0)	l_3	l_5	l_4	a_{16}^4**=0.664**
			l_5	a_{16}^5=0.32
			l_3	a_{16}^3=0.015
x_{17}=(16.0, 14.7, 53.5, 6.8, 3.8, 4.0, 4.0)	l_4	l_5	l_4	a_{17}^4**=0.82**
			l_5	a_{17}^5=0.18
x_{18}=(11.7, 10.0, 42.1, 12.2, 4.3, 5.0, 2.0)	l_3	l_5	l_4	a_{18}^4**=0.936**
			l_5	a_{18}^5=0.036
			l_3	a_{18}^3=0.027
x_{19}=(11.0, 4.2, 60.8, 6.2, 4.8, 4.0, 2.0)	l_3	l_5	l_4	a_{19}^4**=0.8**
			l_3	a_{19}^3=0.165
			l_5	a_{19}^5=0.035
x_{20}=(15.5, 8.5, 56.2, 5.5, 1.8, 4.0, 2.0)	l_3	l_5	l_4	a_{20}^4**=0.846**
			l_5	a_{20}^5=0.093
			l_3	a_{20}^3=0.061
x_{21}=(13.2, 9.1, 74.1, 6.4, 5.0, 2.0, 2.0)	l_2	l_5	l_3	a_{21}^3**=0.64**
			l_4	a_{21}^4=0.342
			l_5	a_{21}^5=0.017
			l_2	a_{21}^2=0.001
x_{22}=(9.1, 4.1, 44.8, 3.3, 10.4, 3.0, 4.0)	l_3	l_5	l_4	a_{22}^4**=0.812**
			l_5	a_{22}^5=0.118
			l_3	a_{22}^3=0.07
x_{23}=(12.9, 1.9, 65.0, 14.0, 7.5, 4.0, 3.0)	l_3	l_5	l_4	a_{23}^4**=0.879**
			l_3	a_{23}^3=0.119
			l_5	a_{23}^5=0.002
x_{24}=(5.9, -27.7, 77.4, 16.6, 12.7, 3.0, 2.0)	l_2	l_4	l_3	a_{24}^4**=0.98**
			l_4	a_{24}^5=0.019
			l_2	a_{24}^3=0.00001
x_{25}=(16.9, 12.4, 60.1, 5.6, 5.6, 3.0, 2.0)	l_3	l_5	l_4	a_{25}^4**=0.702**
			l_3	a_{25}^3=0.258
			l_5	a_{25}^5=0.04

TABLE 8 Cont. "Diagnosis of the health of the firms": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE		SMAA-Classification	
	PES	OPT	Class	Acceptability index
x_{26}=(16.7, 13.1, 73.5, 11.9, 4.1, 2.0, 2.0)	l_3	l_5	l_3	a_{26}^3=**0.66**
			l_4	a_{26}^4=0.334
			l_5	a_{26}^5=0.006
			l_2	a_{26}^2=0.00001
x_{27}=(14.6, 9.7, 59.5, 6.7, 5.6, 2.0, 2.0)	l_3	l_5	l_3	a_{27}^3=**0.51**
			l_4	a_{27}^4=0.471
			l_5	a_{27}^5=0.019
			l_2	a_{27}^2=0.00001
x_{28}=(5.1, 4.9, 28.9, 2.5, 46.0, 2.0, 2.0)	l_1	l_5	l_3	a_{28}^3=**0.635**
			l_4	a_{28}^4=0.292
			l_2	a_{28}^2=0.046
			l_5	a_{28}^5=0.006
			l_1	a_{28}^1=0.001
x_{29}=(24.4, 22.3, 32.8, 3.3, 5.0, 3.0, 4.0)	l_4	l_5	l_5	a_{29}^5=**0.765**
			l_4	a_{29}^4=0.228
			l_3	a_{29}^3=0.007
x_{30}=(29.5, 8.6, 41.8, 5.2, 6.4, 2.0, 3.0)	l_4	l_5	l_4	a_{30}^4=**0.562**
			l_5	a_{30}^5=0.278
			l_3	a_{30}^3=0.16
x_{31}=(7.3, -64.5, 67.5, 30.1, 8.7, 3.0, 3.0)	l_2	l_5	l_3	a_{31}^3=**0.844**
			l_4	a_{31}^4=0.132
			l_2	a_{31}^2=0.023
			l_1	a_{31}^1=0.001
			l_5	a_{31}^5=0.00001
x_{32}=(23.7, 31.9, 63.6, 12.1, 10.2, 3.0, 3.0)	l_3	l_5	l_4	a_{32}^4=**0.875**
			l_3	a_{32}^3=0.122
			l_5	a_{32}^5=0.003
x_{33}=(18.9, 13.5, 74.5, 12.0, 8.4, 3.0, 3.0)	l_3	l_5	l_4	a_{33}^4=**0.626**
			l_3	a_{33}^3=0.372
			l_5	a_{33}^5=0.002
x_{34}=(13.9, 3.3, 78.7, 14.7, 10.1, 2.0, 2.0)	l_2	l_5	l_3	a_{34}^3=**0.873**
			l_4	a_{34}^4=0.122
			l_2	a_{34}^2=0.005
			l_5	a_{34}^5=0.00001
x_{35}=(-13.3, -31.1, 63.0, 21.2, 29.1, 2.0, 1.0)	l_2	l_3	l_2	a_{35}^2=**0.822**
			l_3	a_{35}^3=0.177
			l_1	a_{35}^1=0.001

TABLE 9 Cont. "Diagnosis of the health of the firms": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE		SMAA-Classification	
	PES	OPT	Class	Acceptability index
x_{36}=(6.2, -3.2, 46.1, 4.8, 10.5, 2.0, 1.0)	l_2	l_4	l_3	a_{36}^3**=0.643**
			l_4	a_{36}^4=0.318
			l_2	a_{36}^2=0.028
			l_5	a_{36}^5=0.011
x_{37}=(4.8, -3.3, 71.1, 8.6, 11.6, 2.0, 2.0)	l_3	l_4	l_3	a_{37}^3**=0.871**
			l_4	a_{37}^4=0.128
			l_2	a_{37}^2=0.001
			l_5	a_{37}^5=0.00001
x_{38}=(0.1, -9.6, 42.5, 12.9, 12.4, 1.0, 1.0)	l_2	l_4	l_3	a_{38}^3**=0.678**
			l_2	a_{38}^2=0.212
			l_4	a_{38}^4=0.11
x_{39}=(13.6, 9.1, 76.0, 17.1, 10.3, 1.0, 1.0)	l_2	l_5	l_3	a_{39}^3**=0.692**
			l_2	a_{39}^2=0.245
			l_4	a_{28}^4=0.062
			l_5	a_{28}^5=0.001

TABLE 10 "Credit granting in the banking sector": assignment examples

Reference Alternative x	Class
b_1^1=x_{19}=(13.02, 15.74, 18.02, 7.24, 79.21, 42.63, 79.32)	l_1
b_1^2=x_{23}=(13.66, 11.01, 14.11, 70.55, 69.01, 18.77, 42.39)	l_1
b_1^3=x_{41}=(13.04, 7.99, 22.44, 7.24, 31.4, 14.83, 58.65)	l_1
b_1^4=x_{49}=(13.48, 1.05, 18.02, 6.45, 31.4, 18.77, 100.0)	l_1
b_1^5=x_{68}=(9.91, 7.99, 14.11, 7.24, 12.92, 3.02, 58.65)	l_1
b_1^6=x_5=(14.43, 11.01, 18.02, 29.25, 22.16, 31.73, 39.44)	l_1
b_2^1=x_{29}=(13.39, 11.01, 18.02, 17.36, 22.16, 8.91, 39.44)	l_2
b_2^2=x_{55}=(12.14, 7.99, 18.02, 5.46, 22.16, 3.02, 39.44)	l_2
b_2^3=x_{62}=(11.07, 7.99, 18.02, 5.46, 12.92, 3.02, 39.44)	l_2
b_2^4=x_{66}=(10.25, 7.99, 14.11, 15.58, 12.92, 8.91, 20.24)	l_2
b_3^1=x_{69}=(10.65, 11.01, 10.47, 3.69, 3.76, 8.91, 20.24)	l_3
b_3^2=x_{84}=(8.26, 7.99, 10.47, 3.69, 3.76, 3.02, 20.24)	l_3
b_3^3=x_{94}=(3.86, 1.96, 3.02, 3.69, 3.76, 3.02, 20.24)	l_3

TABLE 11 "Credit granting in the banking sector": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE PES	SMAA-Classification	
		Class	Acceptability index
x_6=(14.15, 11.01, 18.02, 19.13, 49.87, 14.83, 58.65)	l_1	l_1	a_6^1**=0.908**
		l_2	a_6^2=0.092
x_9=(14.54, 14.02, 18.02, 4.65, 12.92, 3.02, 58.65)	l_1	l_1	a_9^1**=0.837**
		l_2	a_9^2=0.161
		l_3	a_9^3=0.002
x_{12}=(14.42, 11.01, 18.02, 4.65, 12.92, 8.91, 58.64)	l_1	l_1	a_{12}^1**=0.832**
		l_2	a_{12}^2=0.15
		l_3	a_{12}^3=0.018
x_{19}=(13.02, 15.74, 18.02, 7.24, 79.21, 42.63, 79.32)	l_1	l_1	a_{19}^1**=1**
x_{21}=(13.6, 4.02, 22.44, 4.65, 31.4, 31.73, 58.65)	l_1	l_1	a_{21}^1**=1**
x_{23}=(13.66, 11.01, 14.11, 70.55, 69.01, 18.77, 42.39)	l_1	l_1	a_{23}^1**=1**
x_{25}=(13.04, 7.99, 22.44, 40.19, 40.64, 6.96, 100.0)	l_1	l_1	a_{25}^1**=0.922**
		l_2	a_{25}^2=0.078
x_{26}=(12.97, 7.99, 18.02, 29.66, 31.4, 18.77, 100.0)	l_1	l_1	a_{26}^1**=0.996**
		l_2	a_{26}^2=0.004
x_{41}=(13.04, 7.99, 22.44, 7.24, 31.4, 14.83, 58.65)	l_1	l_1	a_{41}^1**=1**
x_{49}=(13.48, 1.05, 18.02, 6.45, 31.4, 18.77, 100.0)	l_1	l_1	a_{49}^1**=1**
x_{58}=(10.41, 1.96, 18.02, 19.13, 22.16, 18.77, 100.0)	l_1	l_1	a_{58}^1**=0.922**
		l_2	a_{58}^2=0.074
		l_3	a_{58}^3=0.004
x_{68}=(9.91, 7.99, 14.11, 7.24, 12.92, 3.02, 58.65)	l_1	l_1	a_{68}^1**=1**
x_{73}=(8.65, 1.96, 18.02, 71.48, 59.11, 7.88, 60.12)	l_1	l_1	a_{73}^1**=0.996**
		l_2	a_{73}^2=0.003
		l_3	a_{73}^3=0.001
x_{81}=(7.58, 1.96, 18.02, 71.48, 49.87, 7.88, 60.12)	l_1	l_1	a_{81}^1**=0.996**
		l_2	a_{81}^2=0.004
		l_3	a_{81}^3=0.001
x_{87}=(6.46, 4.98, 18.02, 50.78, 12.92, 6.96, 100.0)	l_1	l_1	a_{87}^1**=0.927**
		l_2	a_{87}^2=0.043
		l_3	a_{87}^3=0.03

TABLE 12 Cont. "Credit granting in the banking sector": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE PES	SMAA-Classification	
		Class	Acceptability index
x_2=(14.64, 14.02, 18.02, 5.46, 22.16, 3.02, 39.44)	l_2	l_2	a_2^2**=0.998**
		l_1	a_2^1=0.001
		l_3	a_2^3=0.001
x_8=(14.71, 14.02, 18.02, 6.43, 12.92, 3.02, 39.44)	l_2	l_2	a_8^2**=0.997**
		l_3	a_8^3=0.02
		l_1	a_8^1=0.01
x_{13}=(14.65, 14.02, 18.02, 6.43, 22.16, 14.83, 39.44)	l_2	l_2	a_{13}^2**=0.904**
		l_1	a_{13}^1=0.095
		l_3	a_{13}^3=0.001
x_{27}=(13.51, 14.02, 18.02, 29.25, 22.16, 8.91, 39.44)	l_2	l_2	a_{27}^2**=0.739**
		l_1	a_{27}^1=0.261
x_{29}=(13.39, 11.01, 18.02, 17.36, 22.16, 8.91, 39.44)	l_2	l_2	a_{29}^2**=1**
x_{37}=(13.52, 11.01, 14.11, 6.43, 22.16, 3.02, 39.44)	l_2	l_2	a_{37}^2**=0.986**
		l_1	a_{37}^1=0.013
		l_3	a_{37}^3=0.001
x_{38}=(13.39, 11.01, 18.02, 5.46, 22.16, 3.02, 39.44)	l_2	l_2	a_{38}^2**=1**
x_{40}=(13.4, 7.99, 14.11, 6.43, 22.16, 8.91, 39.44)	l_2	l_2	a_{40}^2**=0.974**
		l_3	a_{40}^3=0.018
		l_1	a_{40}^1=0.008
x_{48}=(13.24, 4.98, 14.11, 6.43, 22.16, 20.32, 39.44)	l_2	l_2	a_{48}^2**=0.782**
		l_1	a_{48}^1=0.214
		l_3	a_{48}^3=0.004
x_{55}=(12.14, 7.99, 18.02, 5.46, 22.16, 3.02, 39.44)	l_2	l_2	a_{55}^2**=1**
x_{59}=(11.01, 7.99, 18.02, 17.36, 22.16, 3.02, 39.44)	l_2	l_2	a_{59}^2**=0.995**
		l_3	a_{59}^3=0.005
		l_1	a_{59}^1=0.00001
x_{76}=(8.76, 4.98, 14.11, 17.36, 22.16, 3.02, 39.44)	l_2	l_2	a_{76}^2**=0.93**
		l_3	a_{76}^3=0.056
		l_1	a_{76}^1=0.014
x_{88}=(6.65, 4.98, 14.11, 5.46, 3.76, 14.83, 39.44)	l_2	l_2	a_{88}^2**=0.636**
		l_3	a_{88}^3=0.343
		l_1	a_{88}^1=0.021
x_{90}=(4.06, 1.05, 18.02, 39.78, 12.92, 3.02, 39.44)	l_2	l_2	a_{90}^2**=0.554**
		l_1	a_{90}^1=0.319
		l_3	a_{90}^3=0.127
x_{91}=(4.64, 1.96, 6.74, 5.46, 3.76, 3.02, 39.44)	l_2	l_2	a_{91}^2**=0.529**
		l_3	a_{91}^3=0.471
		l_1	a_{91}^1=0.00001

TABLE 13 Cont. "Credit granting in the banking sector": assignments by ELECTRE TRI and SMAA-Classification

Alternative x	ELECTRE PES	SMAA-Classification	
		Class	Acceptability index
x_{66}=(10.25, 7.99, 14.11, 15.58, 12.92, 8.91, 20.24)	l_3	l_3	a_{66}^{3}**=1**
x_{69}=(10.65, 11.01, 10.47, 3.69, 3.76, 8.91, 20.24)	l_3	l_3	a_{69}^{3}**=1**
x_{77}=(9.4, 7.99, 10.47, 3.69, 3.76, 3.02, 20.24)	l_3	l_3	a_{77}^{3}**=1**
x_{84}=(8.26, 7.99, 10.47, 3.69, 3.76, 8.91, 20.24)	l_3	l_3	a_{84}^{3}**=1**
x_{85}=(8.45, 7.99, 6.74, 3.69, 3.76, 8.91, 20.24)	l_3	l_3	a_{85}^{3}**=1**
x_{92}=(5.11, 4.98, 3.02, 3.69, 3.76, 3.02, 20.24)	l_3	l_3	a_{92}^{3}**=1**
x_{93}=(3.68, 1.96, 6.74, 15.58, 3.76, 3.02, 20.24)	l_3	l_3	a_{93}^{3}**=1**
x_{94}=(3.86, 1.96, 3.02, 3.69, 3.76, 3.02, 20.24)	l_3	l_3	a_{94}^{3}**=1**
x_{95}=(2.56, 1.96, 6.74, 15.58, 3.76, 14.83, 20.24)	l_3	l_3	a_{95}^{3}**=1**
x_{96}=(2.81, 1.96, 3.02, 3.69, 5.41, 3.02, 20.24)	l_3	l_3	a_{96}^{3}**=1**
x_{97}=(1.04, 1.05, 14.11, 49.9, 3.76, 3.02, 20.24)	l_3	l_3	a_{97}^{3}**=0.755**
		l_1	a_{97}^{1}=0.242
		l_2	a_{97}^{2}=0.003
x_{98}=(1.48, 1.05, 3.02, 15.58, 3.76, 3.02, 20.24)	l_3	l_3	a_{98}^{3}**=1**
x_{99}=(1.68, 1.96, 3.02, 15.58, 5.41, 3.02, 20.24)	l_3	l_3	a_{99}^{3}**=1**
x_{100}=(0.6, 1.05, 0.71, 3.69, 5.41, 3.02, 20.24)	l_3	l_3	a_{100}^{3}**=1**

TABLE 14 "Identification of accident type": assignment examples

Reference Alternative x	Class
b_1^1=(11, 17, 15)	l_1
b_1^2=(11, 18, 11)	l_1
b_2^1=(10, 13, 18)	l_2
b_3^1=(15, 14, 12)	l_3
b_3^2=(15, 16, 11)	l_3
b_3^3=(16, 15, 14)	l_3

TABLE 15 "Identification of accident type": assignments by TRINOMFC and SMAA-Classification

Alternative x	TRINOMFC		SMAA-Classification	
	Class	Similarity index	Class	Acceptability index
x_0=(16, 14, 14)	l_3	SI_0^3**=0.79**	l_3	a_0^3**=1**
	l_2	SI_0^2=0.48	l_2	a_0^2=0
x_1=(10, 18, 11)	l_1	SI_1^1**=0.7**	l_1	a_1^1**=0.991**
	l_3	SI_1^3=0.21	l_3	a_1^3=0
	l_2	SI_1^2=0.1	l_2	a_1^2=0.009
x_2=(15, 17, 18)	l_2	SI_2^2**=0.72**	l_2	a_2^2**=0.17**
	l_3	SI_2^3=0.65	l_3	a_2^3=0.54
	l_1	SI_2^1=0.3	l_1	a_2^1=0.29

6.1.2 Concluding Remarks

Let us, finally, emphasize that even though the new SMAA-Classification method does not require a great deal of parametric information from the DM (such as thresholds and/or weights) and, thus, it does not load him or her cognitively, still it provides similar results to methods that assume that the DM is able to define all the parameters precisely. Thus, SMAA-Classification is more flexible while being as effective.

We have tested our method with the results obtained by applying ELECTRE TRI on two different data sets and by classifying with TRINOMFC. The results of such experiments speak for the effectiveness of the SMAA-Classification method.

6.2 Dichotomic Classification

In Section 4.2, we have introduced the interactive Dichotomic Classification (DC) method for ordinal classification of a set of given alternatives into a predefined number of classes. The method assumes that the set of alternatives to be classified is given a priori and it is estimated on the set of criteria that may have verbal or numerical values. The number of classes and their order should also be available in advance. The Dichotomic Classification method utilizes the ideas of VDA classification methods. In the method, the DM should assign to classes some selected alternatives from the set of given alternatives, but not all. In order to select the alternative to be posed to the DM at each iteration of the classification, the Dichotomic Classification method adopts the bisection search.

We have also considered the possibility of utilizing information about the lexicographic order of the criteria according to their importance to the DM, if it is available. The procedure of classification is simplified and the number of questions posed to the DM, naturally, is reduced if such a relation can be applied.

Next, we consider some numerical tests of the developed method and compare the results provided by the Dichotomic Classification method and the SAC method on the same artificial data sets.

6.2.1 Analysis of Results

The effectiveness of the VDA methods can be estimated as a number of questions posed to the DM. Thus, in order to test performance of the Dichotomic Classification method and to compare it to the VDA methods developed earlier, we have to estimated the number of questions posed to the DM for different classification problems.

When compared to the VDA methods developed earlier, the Dichotomic Classification method can work for the set of given alternatives without using the complete set obtained as a Cartesian product of all criteria scales. However, the DM should be questioned for the classification of each such new set of given alternatives. The exception is the case when the set of given alternatives is equal to the complete set of alternatives obtained as a Cartesian product of all criteria values (that is an universal set of alternatives for the selected set of criteria). Then, we can search for the rules that describe classes with boundary alternatives in a similar way it is done in the ORCLASS and SAC methods. Then, there is no need of questioning the DM each time the new set of alternatives (that is smaller than the Cartesian product of criteria scales) has to be classified. Boundary alternatives can be detected when all the alternatives have been classified with regards to the property, according to which, for the boundary alternative, changing the value on at least one criterion moves it to a neighboring class. Here, we use this exceptional case in order to compare Dichotomic Classification and SAC methods.

An approach to the comparison of VDA methods has been proposed in [93]. Theoretically, the minimal number of questions posed to a DM in the preference elicitation procedure would concern only the alternatives that lie on the boundaries, but that would be an ideal case and does not work in practice. However, the efficiency of the preference elicitation procedure can be estimated with the help of this ideal case. The relation between the number of boundary alternatives and the number of questions posed to the DM for a complete classification is often referred to as the *absolute effectiveness index* for classification methods. The number of alternatives to be classified by the DM is known as the *relative effectiveness index* [93].

We introduce another *reducing effectiveness index* (E) as a relation of the number of questions (Q) posed to the DM and the number of given alternatives (m): $E = Q/m$; $E \in [0, 1]$. This index shows the reduction of the number of alternatives classified by the DM when compared to the direct classification of every alternative in X. In the worst case, when the set of questions asked is equal to the set of given alternatives, the reducing effectiveness is equal to 1. That is why we are trying to minimize the reducing effectiveness index.

The absolute and reducing effectiveness indices are the opposites in the following sense. In the "optimistic" absolute index, the number of questions posed to the DM is compared to the best or ideal case and we should maximize it, while in the "pessimistic" reducing index, the number of questions is compared to the worst case of the preference elicitation procedure - to direct classification of each alternative from the set of given alternatives. However, for the absolute index, the information about exact boundaries should be available. That is not always the case, especially, when the set of given alternatives is not equal to the Cartesian product and the boundary alternatives detected by some method do not represent any compact rule [92]. Thus, in the following we compare SAC and Dichotomic Classification according to the relative and reducing indices.

On the other hand, in Dichotomic Classification method, we have tried to avoid embedded heuristics (about uniform distribution of the centers of unknown classes) of the other VDA methods and create a method that works efficiently for different distributions of alternatives between classes. The method introduced should be effective for different kinds of alternatives distributions. Real life examples studied in [114] motivate us to estimate the performance of methods for different alternatives distributions between classes such as uniform (when the same (+/-2) amount of alternatives are allocated in each class), exponential (when the number of alternatives in the next class is larger than in the previous one), and Gaussian (when almost all alternatives are assigned into one class). We have also generated random alternatives distributions.

Another direction of research examines the DMs' behavior when determining boundaries. The experiments conducted in [96] show that boundary alternatives have a certain structure. Usually, experts and/or DMs develop a small number of rules for the classification of alternatives. Each rule has a structure of a tree with the combination of the most important criterion in the root, to which several criteria that are necessary to make a decision are added. Thus, boundaries should be simulated according to the structure described. These experiments conducted in [96] confirm our observations of the DMs' behavior and directs us to the idea that for the DM the criteria may have different importance. There are at least two possible cases: either the DM assigns some weights to each criterion or simply orders them lexicographically. In both cases, the task of classification is significantly simplified. It is challenging to use weights in VDA where verbal values are utilized, that is why we have used the order on the set of criteria. The information about the lexicographical order is applied to the studied and developed methods and their performance is estimated and compared.

Besides the effectiveness of the methods, it is important to evaluate their computational complexity, even though the technologies are improving and complex calculations are becoming less time-consuming. In VDA type of methods, the computational complexity of a method can be calculated as the number of comparison operations that the algorithm should perform in order to select the alternative that is proposed to the DM for classification at each iteration of the preference elicitation procedure. In the SAC method, the alternative is chosen according to informativeness, for the calculation of which the preference relation should be checked for each alternative with regards to each not yet classified alternative and each available class. Thus, the computational complexity at each iteration of the SAC method is $m' \times m' \times l$, where m' is the number of not yet classified alternatives and l is the number of available classes for this alternative. Unlike in SAC, in the Dichotomic Classification method for the selection of the alternative that will be proposed to the DM for direct classification, we need only information about the number of not yet classified alternatives m' at each iteration. At the first iteration, the number of the not yet classified alternatives is equal to the complete set of given alternatives $m' = m$, but it decreases from iteration to iteration as some of the alternatives are classified. Next, we demonstrate the results of tests with the SAC and Dichotomic Classification methods with equally important criteria and with a lexicographic order on the set of criteria, and evaluate them according to the effectiveness criteria for the problems with different alternatives distributions.

For our tests, we select problems with the following parameters: 4, 5 or 6 criteria and 3 values on each criterion. Resulting from Cartesian products we have the sets of 81, 243 and 729 alternatives that should be assigned into 2, 3 or 4 classes. We also consider alternative sets with 256, 1024 and 4096 alternatives that are created as Cartesian products of 4, 5 or 6 criteria, each of which is estimated on 4 values and the alternatives are classified into 2, 3 or 4 classes.

Statistical modeling of different alternatives distributions between classes allows comparing the effectiveness of the SAC and Dichotomic Classification methods with a lexicographic order on the set of criteria and without it, according to the efficiency indices measuring average relative effectiveness and reducing effectiveness. We perform all the experiments with artificial data sets. For problems with different sizes, we produce from 5 to 10 simulations for each type of distribution. For simulations according to the uniform, exponential and Gaussian distributions, we perform 7 and 5 runs for 3 and 4 values on the scales of criteria, respectively, and round the calculated average to the closest integer. For random distribution, we perform the same simulations, however, without rounding the relative effectiveness.

The relative effectiveness of SAC and Dichotomic Classification estimated as a number of questions posed to the DM for direct classification is presented in Table 16 and Table 17 (for 3 and 4 values on the scales of criteria, respectively). Here, the first columns show the number of criteria (the size of the set of criteria G), the second columns define the number of alternatives, that is, a Cartesian product of the criteria set (the size of the set of alternatives X), and the third columns present the number of classes (the size of the sets of classes L). The fourth columns define the method (Dichotomic Classification or SAC). For each type of problem, we use the methods with the following alternatives distributions: uniform, exponential, random and Gaussian, which are presented in columns five to eight, respectively. At the row intersections of the columns five to eight, the numbers of questions posed to the DM are located. The cells highlighted in italic face indicate the cases in which SAC and Dichotomic Classification are equally effective or when SAC works better. The tables also contain statistical information about averaging of the relative effectiveness for all distributions and standard deviation in columns nine and ten, respectively.

TABLE 16 Relative effectiveness of DC and SAC for 3 value scales

Criteria	Alternatives	Classes	Method	Uniform Distr.	Expon. Distr.	Random Distr.	Gauss. Distr.	Average Distr.	Stand. Dev.
4	81	2	DC	20.00	16.00	18.00	10.00	16.00	5.28
			SAC	30.00	37.00	22.00	49.00	34.50	10.97
5	243	2	DC	35.00	37.00	46.66	21.00	34.91	11.96
			SAC	92.00	298.00	75.66	146.00	152.91	90.88
6	729	2	DC	66.00	70.00	81.33	56.00	68.33	16.60
			SAC	98.00	332.00	149.66	434.00	253.41	132.45
4	81	3	DC	23.00	29.00	24.66	32.00	27.16	4.27
			SAC	25.00	33.00	26.66	55.00	34.75	11.80
5	243	3	DC	42.00	49.00	48.66	55.00	48.66	5.13
			SAC	30.00	61.00	65.33	82.00	59.58	21.29
6	729	3	DC	88.00	94.00	140.66	135.00	114.41	25.45
			SAC	33.00	80.00	175.33	241.00	132.33	76.12
4	81	3	DC	*38.00*	25.00	24.00	32.00	29.75	5.88
			SAC	*42.00*	40.00	25.00	56.00	40.75	13.18
5	243	3	DC	*73.00*	62.00	57.00	55.00	61.75	7.90
			SAC	*66.00*	62.00	61.00	72.00	65.25	6.28
6	729	3	DC	*148.00*	128.00	140.66	135.00	137.91	9.90
			SAC	*132.00*	101.00	109.00	151.00	123.25	37.80

In Figures 18- 21 and in Figures 22- 25, we demonstrate the comparative results of the relative effectiveness of SAC and Dichotomic Classification with different alternatives distributions. Here, the horizontal axes represent the number of classes and the number of alternatives for each problem; the vertical axes show the relative effectiveness that is estimated as a number of questions posed to the DM. The first group of figures follows the behavior of relative effectiveness depending on the size of the criteria set: we fix the number of classes and change the number of the criteria. The second group tracks changing the size of classes set: we fix the number of criteria and change the number of classes.

According to the results, the Dichotomic Classification method is more effective in most cases of exponential and all cases of Gaussian distribution, while when the alternatives are distributed uniformly at some cases SAC works in the same way as (or better than) Dichotomic Classification. For a random distribution, Dichotomic Classification performs better on the average, however, there are cases in which SAC and Dichotomic Classification are equally effective. The analysis of the boundaries of these random distributions show that in such cases the distribution is close to a uniform one.

The effectiveness of SAC for the uniform distribution, when compared with

TABLE 17 Relative effectiveness of DC and SAC for 4 value scales

Criteria	Alternatives	Classes	Method	Uniform Distr.	Expon. Distr.	Random Distr.	Gauss. Distr.	Average Distr.	Stand. Dev.
4	256	2	DC	13.00	17.00	24.00	15.00	17.25	9.41
			SAC	104.00	104.00	28.66	149.00	96.41	41.96
5	1024	2	DC	17.00	30.00	69.33	20.00	34.08	39.17
			SAC	358.00	396.00	172.00	516.00	360.50	147.21
6	4096	2	DC	21.00	56.00	156.33	25.00	64.58	39.17
			SAC	510.00	482.00	538.33	550.00	520.08	30.44
4	256	3	DC	44.00	32.00	46.66	60.00	45.50	9.68
			SAC	77.00	60.00	63.33	113.00	78.33	20.38
5	1024	3	DC	97.00	48.00	106.33	165.00	104.08	39.41
			SAC	114.00	118.00	150.33	243.00	156.33	79.40
6	4096	3	DC	249.00	56.00	271.66	165.00	185.25	39.41
			SAC	252.00	453.00	585.33	622.00	478.08	167.29
4	256	3	DC	*45.00*	34.00	47.33	60.00	46.58	10.69
			SAC	*39.00*	49.00	65.22	151.00	76.08	40.10
5	1024	3	DC	*103.00*	59.00	148.33	165.00	118.83	44.05
			SAC	*84.00*	126.00	224.33	199.00	158.33	61.06
6	4096	3	DC	*563.00*	132.00	239.33	558.00	373.08	44.06
			SAC	*523.00*	488.00	588.33	702.00	575.33	94.13

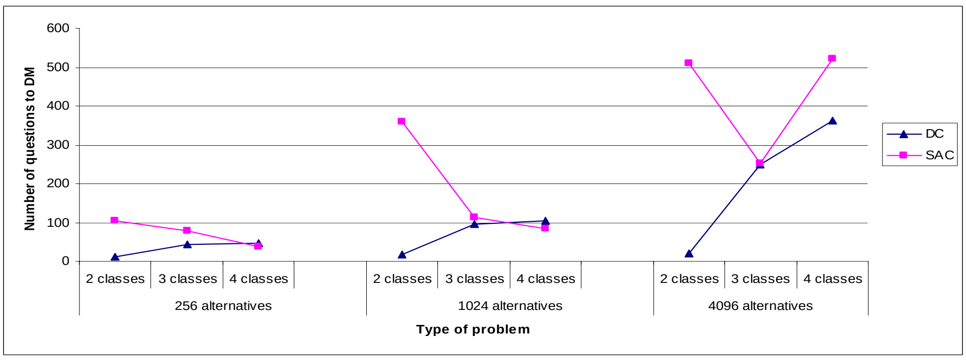

FIGURE 18 Uniform distribution of alternatives between classes (grouped by criteria)

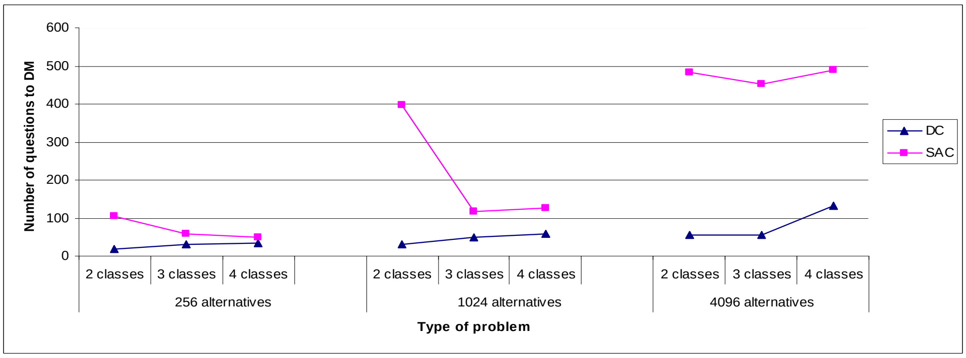

FIGURE 19 Exponential distribution of alternatives between classes (grouped by criteria)

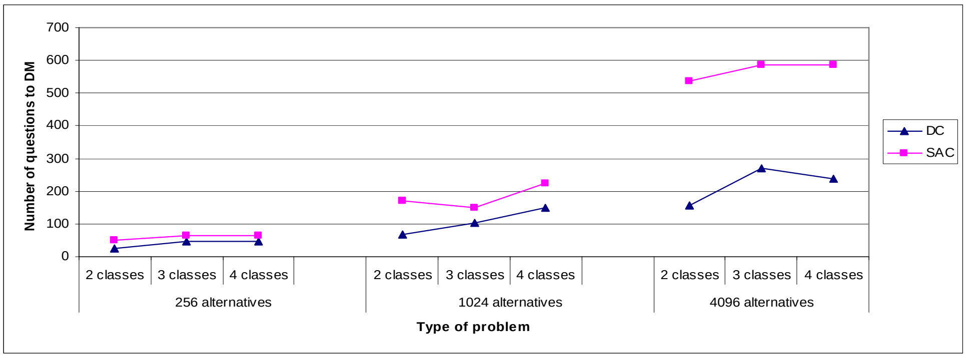

FIGURE 20 Random distribution of alternatives between classes (grouped by criteria)

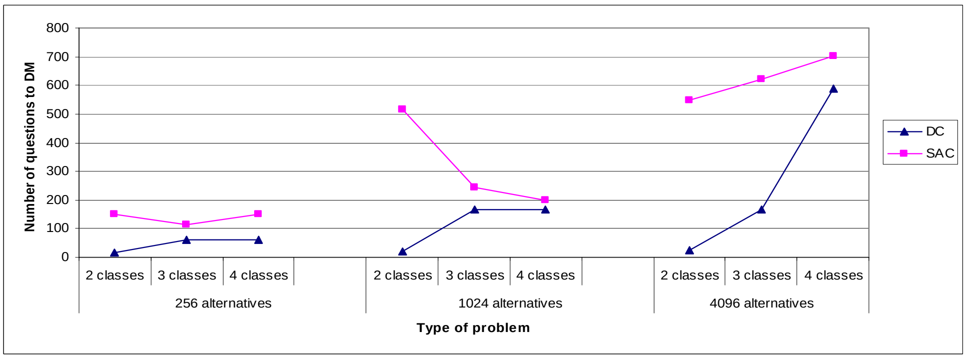

FIGURE 21 Gaussian distribution of alternatives between classes (grouped by criteria)

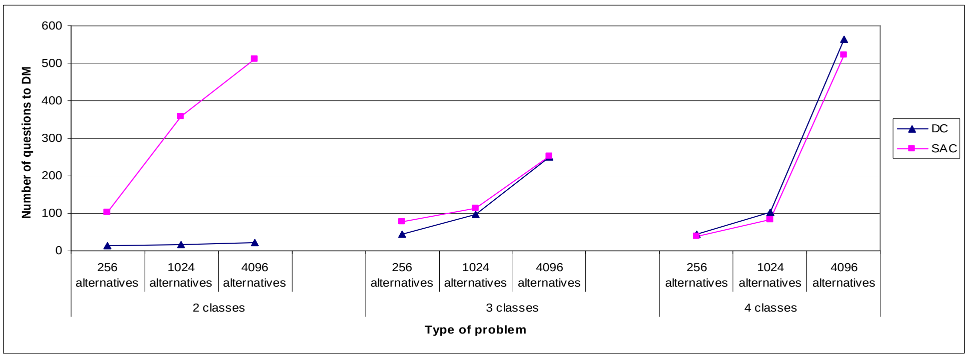

FIGURE 22 Uniform distribution of alternatives between classes (grouped by classes)

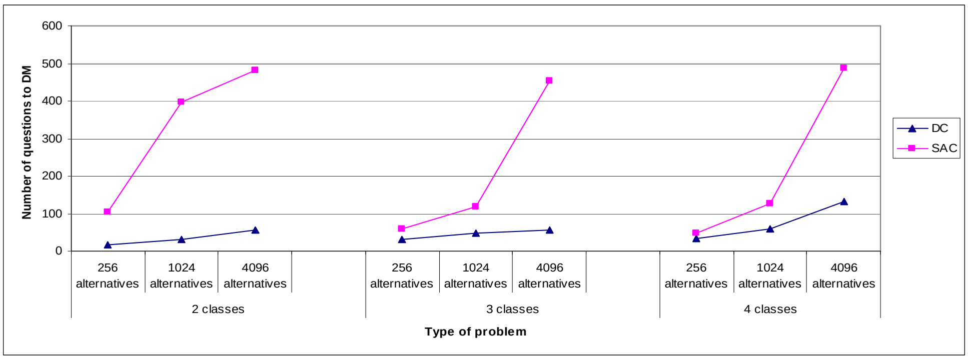

FIGURE 23 Exponential distribution of alternatives between classes (grouped by classes)

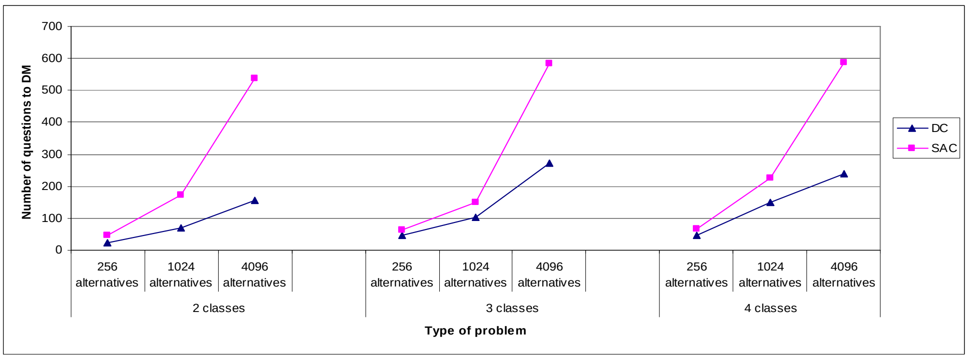

FIGURE 24 Random distribution of alternatives between classes (grouped by classes)

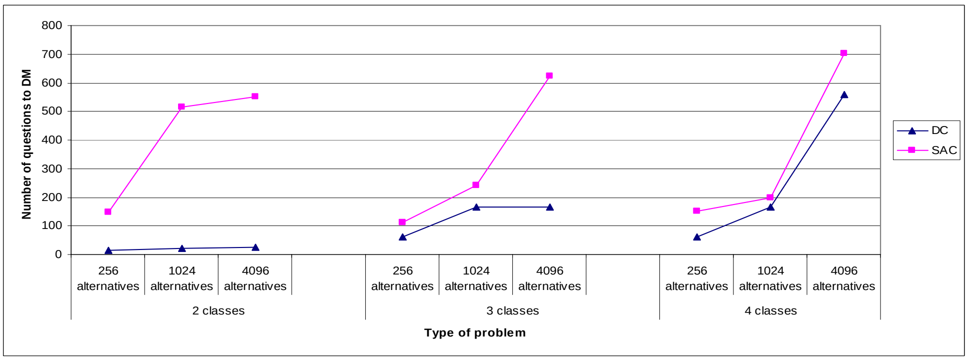

FIGURE 25 Gaussian distribution of alternatives between classes (grouped by classes)

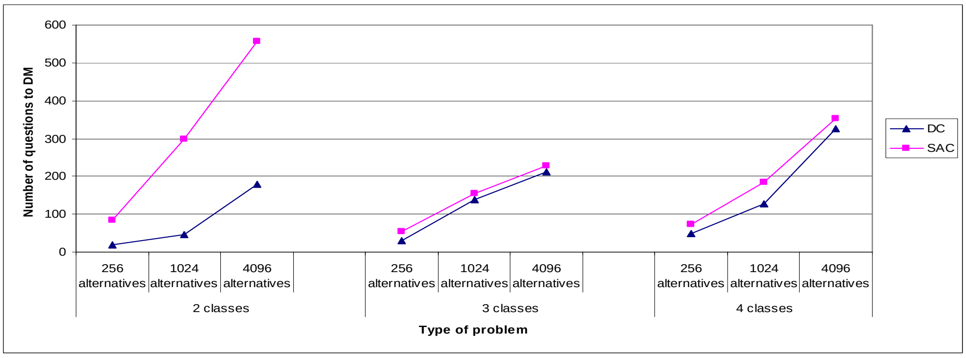

FIGURE 26 Average relative effectiveness of DC and SAC methods (grouped by classes)

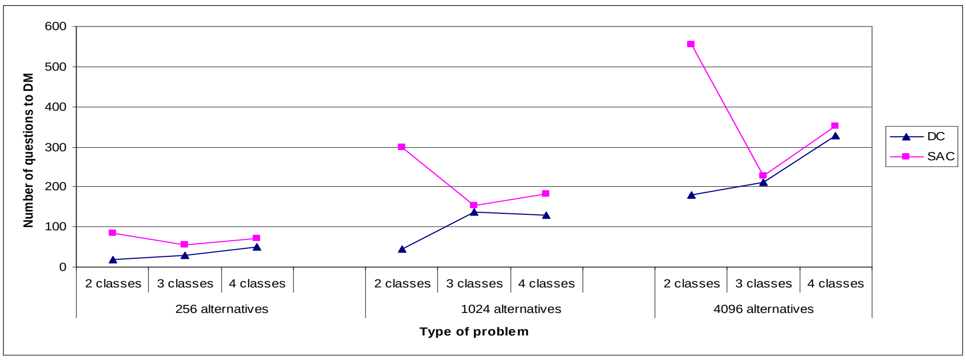

FIGURE 27 Average relative effectiveness of DC and SAC methods (grouped by criteria)

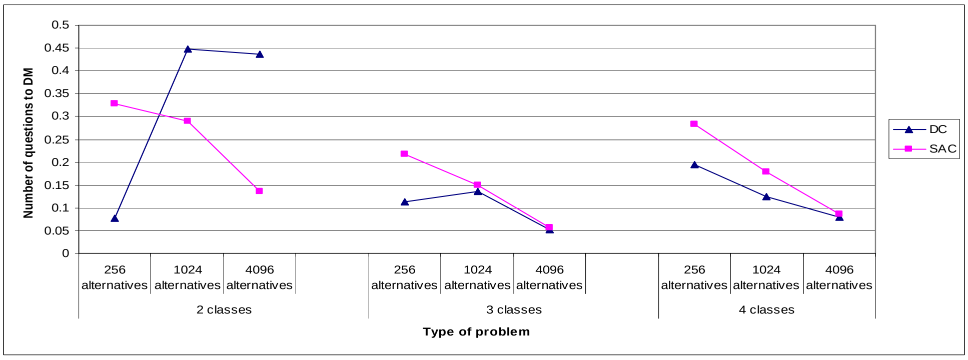

FIGURE 28 Reducing effectiveness of DC and SAC methods (grouped by classes)

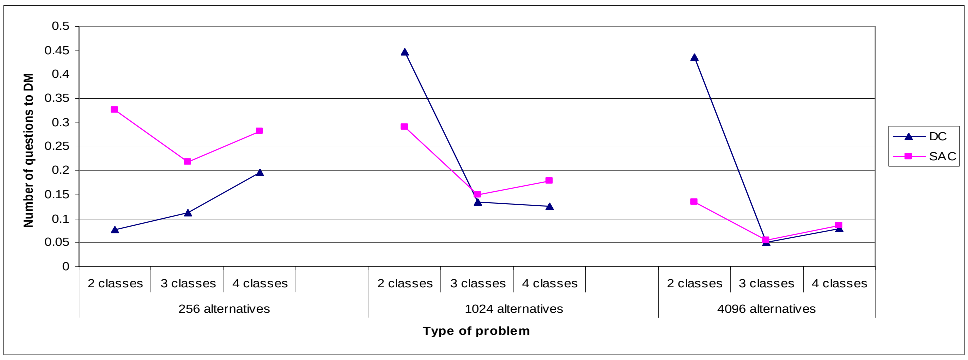

FIGURE 29 Reducing effectiveness of DC and SAC methods (grouped by criteria)

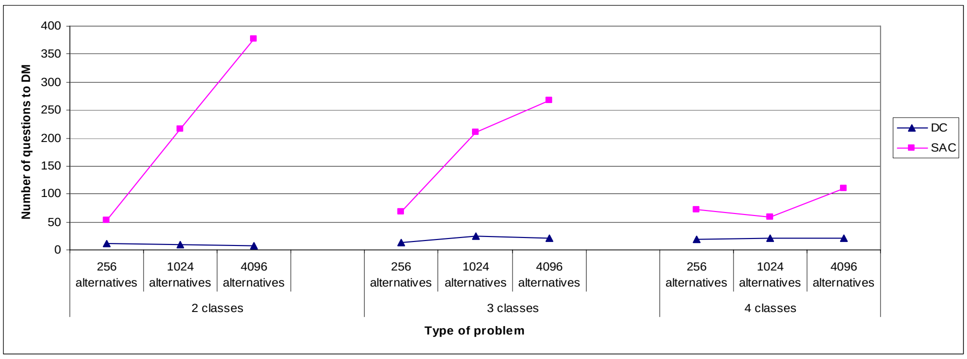

FIGURE 30 Average relative effectiveness of DC and SAC methods with the lexicographic ordering of criteria (grouped by classes)

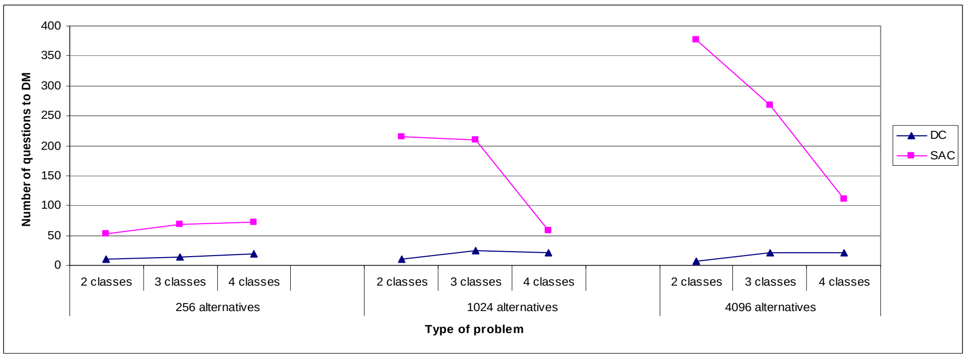

FIGURE 31 Average relative effectiveness of DC and SAC methods with the lexicographic ordering of criteria (grouped by criteria)

the other distributions, is caused by the heuristics embedded into the method. When calculating the probability of each alternative to be assigned in a class, it is assumed that the center of an unknown class is located with an equal distance from the centers of already known classes, in other words, the centers of classes are uniformly distributed. This is also the reason for a difference between the values of SAC's average relative effectiveness presented in [95] and the values obtained in this work.

Figures 26 and 27 show the relative effectiveness averaged for all distributions. As in the previous figures, the types of problems are grouped by the same criterion in Figure 26 and by the same class in Figure 27. These figures illustrate a dependence between the number of criteria (or classes) and the number of questions posed to the DM. Thus, the average relative effectiveness increases when the number of criteria increases for both SAC and Dichotomic Classification and reduces for SAC and increases for Dichotomic Classification when the number of classes grows. The same fact is confirmed by the reducing effectiveness index presented in Figures 28 and 29. On an average, the number of questions asked from the DM by Dichotomic Classification is smaller than by SAC, and according to the standard deviation the method has a smaller variation in this number.

The data presented in Table 18 shows the performance of SAC and Dichotomic Classification for different distributions of alternatives between classes when a lexicographic order on the set of criteria is considered. Statistical analysis of the table shows that the Dichotomic Classification method works less effectively for some cases of the uniform and random distributions (the cells are highlighted in italic face); however, for the exponential and Gaussian distributions it performs better. Figures 30 and 31 demonstrate the average relative effectiveness of the SAC and Dichotomic Classification methods when the lexicographic ordering of criteria is considered. The problem types are ordered in Figures 30 and 31 by criteria and classes, respectively. According to the average relative and reducing effectiveness indices, Dichotomic Classification is more effective than SAC for the problems with a lexicographically ordered set of criteria.

The tests described and both the relative and reducing effectiveness indices show better results on the average for the Dichotomic Classification method compared to the SAC method on the same data sets for both cases: with the lexicographical order between criteria and without such information. However, a more detailed analysis of the different alternatives distributions between classes demonstrates that for the uniform distribution SAC performs as well as Dichotomic Classification and even better in some cases (see Figure 22, for instance). That is due to the specificity of the algorithm of the method. We also observe increasing effectiveness of SAC with an increasing number of classes, while the effectiveness of Dichotomic Classification decreases in the same conditions (see Figure 28, for example).

TABLE 18 Relative effectiveness of Dichotomic Classification and SAC for 4 value scales when criteria are lexicographically ordered

Criteria	Alternatives	Classes	Method	Uniform Distr.	Expon. Distr.	Random Distr.	Gauss. Distr.	Average Distr.	Stand. Dev.
4	256	2	DC	*12.00*	12.00	*11.66*	7.00	10.66	2.45
			SAC	*2.00*	78.00	*2.00*	127.00	52.25	61.37
5	1024	2	DC	*10.00*	10.00	*10.00*	9.00	9.75	0.50
			SAC	*2.00*	314.00	*4.50*	541.00	215.37	261.88
6	4096	2	DC	*8.00*	8.00	*8.00* 7.00	7.75	0.50	
			SAC	*2.00*	693.00	*2.66*	812.00	377.41	435.82
4	256	3	DC	15.00	14.00	*9.66*	17.00	13.91	3.10
			SAC	43.00	62.00	*5.83*	161.00	67.96	66.26
5	1024	3	DC	19.00	48.00	*14.17*	19.00	25.04	15.47
			SAC	172.00	226.00	*9.00*	431.00	209.50	171.10
6	4096	3	DC	23.00	23.00	*15.17*	22.00	20.80	3.77
			SAC	210.00	332.00	*8.00*	519.00	267.25	214.52
4	256	3	DC	21.00	20.00	*13.33*	20.00	18.58	3.53
			SAC	21.00	68.00	*6.66*	190.00	71.41	83.28
5	1024	3	DC	27.00	25.00	*15.00*	20.00	21.75	5.37
			SAC	69.00	73.00	*11.66*	159.00	78.16	60.73
6	4096	3	DC	33.00	32.00	*24.17*	22.00	27.79	5.52
			SAC	110.00	89.00	*8.33*	236.00	110.83	94.25

6.2.2 Concluding Remarks

We have implemented the Dichotomic Classification method and performed tests on artificial data sets. When compared to another VDA classification method, SAC, the Dichotomic Classification method is more effective as attested by the relative and reducing indices of effectiveness. Dichotomic Classification is also computationally less complex. Improvement in the effectiveness and computational complexity allows overcoming the limits of the SAC method for the set of alternatives classified. The Dichotomic Classification method works better for different alternatives distributions between classes when compared to SAC except for uniform distribution, where in some cases they both performed equally well or SAC outperformed Dichotomic Classification.

In the VDA methods developed earlier, the heuristics about distribution of alternatives uniformly between classes has been assumed. However, in the real-life, it is not always the case. For instance, the real-life examples presented in [114] show that there may be different distributions of alternatives between classes. That is why, in this work, we have considered artificial data sets with different distributions of alternatives between classes. The effective work of Dichotomic Classification for different distributions of alternatives between classes is confirmed by the tests above.

We also considered the possibility of establishing a lexicographic order on the set of criteria. In case when such information is available, the classification procedure is simplified and the number of questions posed to the DM for direct classification is reduced. In the tests, Dichotomic Classification performed better than SAC on the average also in this setting.

7 APPLICATION OF DICHOTOMIC CLASSIFICATION TO ADHD DIAGNOSTICS

An important application area of VDA methods for classification is diagnostics of alternatives that are usually estimated with verbal or numerical values. Let us mention that the VDA method ORCLASS has been successfully applied in medical diagnostics [99]. As one of many possible applications, we consider here neuropsychological diagnostics, where the evaluation of a patient's behavior is usually performed in a verbal form and is sensitive to any transformation. Our special interest is focused on aiding decision making when detecting different disorders. Next, we present the classification model developed for the diagnostics and identification of the Attention Deficit - Hyperactivity disorder (ADHD). This work has been done in cooperation with Niilo Mäki Institute (Jyväskylä), where there is a wealth of experience in solving such problems; however, the need of a computerized multicriteria decision support system exists. The work on the multicriteria decision support system is discussed with experts in neuropsychological diagnostics and is to be improved according to their needs. The system can assist an expert as well as a new specialist in neuropsychological diagnostics.

7.1 Introduction to ADHD Diagnostics

In neuropsychology, there are a number of disorders that are difficult to identify and for which diagnostic criteria are not completely defined. ADHD is one of the most common and the most extensively studied child psychiatric syndromes [150]. This disorder is characterized by symptoms of inattention, hyperactivity and impulsivity [2]. Major research of ADHD is based on the studies made with children of the school age between 6 and 12 years. Even though some symptoms may appear at a younger age, it is difficult to estimate whether a child's behavior differs from the normal one. At the school age, the child needs to exercise concentration and vigilance; he or she should also demonstrate the abilities of motor control and working memory [77]. Although most of the research is done on young ADHD children, several investigations on adults [134] indicate developmental problems during the lifespan of persons affected by ADHD. Thus, ADHD can cause various negative consequences including poor achievement at school, more visits to the emergency rooms and more automobile accidents [134].

Currently, diagnostics of ADHD is performed with a standard questionnaire known as Structured Interview for Diagnostic Assessment of Children (SIDAC) for Diagnostic and Statistical Manual for Mental Disorders (DSM). However, this standard is constantly modified and has already undergone several revisions (DSM-I, ..., DSM-IV) [2], [54]. Thus, ADHD diagnostics is not yet even properly defined and cannot be objectively measured [28]. In this work, we consider the latest version of SIDAC for DSM: DSM-IV. The last version of SIDAC for DSM-IV diagnoses for children of ages 6-12 can be found in [2], [54].

Another feature of the ADHD diagnostics is the descriptive character of the symptoms that allow estimating the behavior of a person (in this work, we assume the person under study to be a child). The usual way of diagnostics is based on the evaluation of the child's behavior by parents and teachers as well as on the child's own reports. These data should be enough for a clinician to be able to detect the existence of one of ADHD subtypes or conclude the absence of ADHD. In the ADHD diagnostics, the DM is a clinician.

Diagnostic problems, including ADHD detection, are typical classification tasks where it is necessary to assign a set of alternatives estimated on a set of criteria into a predefined set of classes. The above-mentioned features of the ADHD diagnostic problems, among them are poorly or subjectively measured nature of symptoms as well as the descriptive or verbal manner of criteria estimations, narrow down the set of possible methods that can be applied to resolve such classification tasks. In particular, difficulties with precise definition of numerical estimations when describing behavior should be taken into account. That is why we have applied the Dichotomic Classification method in the ADHD diagnostics [162]. Next, we briefly describe the ADHD problem area in more detail.

As mentioned before, the ADHD diagnostics is performed with a SIDAC interview. The structured interview assumes that all participating persons answer diagnostic questions about the behavior of the child. Usually, the following individuals are involved in the questioning procedure: the parents (usually, the mother), the teacher, and the child itself; the last modifications to DSM-IV also include the clinician's opinion about the behavior of the child when the diagnosis is made. There are different opinions about the importance of the answers of different persons. Some researchers point out that the teacher's opinions are the most important; the others do not place similar importance to this aspect [28]. We assume that the opinions of the mother and the teacher are the most important ones; however, the child's self-reporting is also taken into account, and we include the clinician on the stage of decision making. In this, we follow the current practice in ADHD diagnostics.

The structured interview for ADHD in SIDAC for DSM-IV [2] is presented in Appendix. ADHD consists of three groups of symptoms: inattention, impulsivity and hyperactivity. The decision is to be made about the presence of one of the three following subtypes of ADHD: 1) attention-deficit hyperactivity disorder with predominantly inattentive type (inattention), 2) attention-deficit hyperactivity disorder with predominantly hyperactive-impulsive type (hyperactivity-impulsivity), 3) attention-deficit hyperactivity disorder: combined type (ADHD) or the absence of any of these disorders (no ADHD). The decision is based on the reports of the three persons involved in the decision making. The whole questionnaire contains eighteen questions. The first nine questions allow making conclusions about the presence or absence of inattention. The second part of the interview consists of other nine questions that show whether there is the hyperactivity-impulsivity disorder. The conclusion about inattention is made when the child has shown six of the inattention symptoms and less than six of the hyperactivity-impulsivity symptoms. The hyperactivity-impulsivity disorder is detected when the child's behavior accords with six symptoms from the hyperactivity-impulsivity group and with less than six of the inattention group. For the ADHD, six symptoms from the first and six from the second group should be satisfied. Otherwise, the conclusion about the absence of the ADHD is made. These three rules are used in order to imitate the behavior of a clinician when making a decision about the presence of the disorder discussed here.

Let us now model the problem of ADHD diagnostics with multiple criteria decision analysis and illustrate how the Dichotomic Classification method can assist in ADHD diagnostics.

7.2 ADHD Diagnostics with Dichotomic Classification

7.2.1 ADHD Model for Classification

In order to see how the Dichotomic Classification method works for the ADHD diagnostics, let us present the problem in multiple criteria analysis terms. We consider diagnostics of inattention and hyperactivity-impulsivity separately because the symptoms for these two diagnoses are different and they do not interact with each other (see Appendix). The conclusion about ADHD or the absence of ADHD is made based on the combination of the results from the inattention and the hyperactivity-impulsivity diagnostics.

Thus, we will have three stages in our diagnostics: 1) diagnostics of inattention with two possible classes: either the child meets the diagnostic criteria or not; 2) diagnostics of hyperactivity-impulsivity with two possible classes: either the child meets these diagnostic criteria or not; and 3) combination of the results of two previous stages with four possible classes for ADHD diagnostics: the child meets the diagnostic criteria for both of the previous stages, one of two or none of them.

We formulate the first two stages as separate classification problems and then combine their results for the final ADHD diagnostics decision at the third stage. The criteria for the first and second stages of the diagnostics consist of questions about the behavior of the child for detecting inattention and hyperactivity-impulsivity. The values on each criterion are the same and they reflect the answers of the mother, the teacher and child him/herself. Even though we assume the answers of the mother and the teacher to be more important than the child's self-reporting, the overall estimation of child's behavior is considered by a clinician based on all information available. Thus, for each of the nine symptoms (criteria) we compare the answers obtained from the three persons about the behavior of the child (criteria scale): if the majority of the interviewers (2 or 3 out of 3) agree that a symptom appears often, we assume that the symptom exists, otherwise if only a minority (0 or 1 from 3) agrees that a symptom appears often, we assume that the symptom does not exist.

In all, the ADHD diagnostics according to the SIDAC for DSM-IV diagnoses of children of ages 6-12 (see Appendix) is done in three stages. At the first stage, a set of nine questions with two possible answers (rarely or often) on each of them allows the creation of the set of all possible combinations (Cartesian product) of answers, that is, the set of alternatives. This set is classified into two classes: inattention exists or inattention does not exist. The second stage is done by an analogy with the first one with the only difference that another set of nine questions is taken for evaluation regarding whether the child meets with the hyperactivity-impulsivity conditions or not. At the third stage of the diagnostics, we aggregate the results of the two previous stages and obtain the evaluation for the ADHD: whether the child meets both subtypes of criteria (inattention and hyperactivity-impulsivity), or one of them (only inattention or only hyperactivity-impulsivity), or does not have any of the ADHD subtypes.

7.2.2 Dichotomic Classification for ADHD Diagnostics

We demonstrate the Dichotomic Classification method with the first stage of decision aiding for ADHD diagnostics where the existence or absence of inattention is detected. In Appendix, the nine questions for the identification of inattention are presented. For example, the first question item in this interview is: "Often fails to give close attention to details or makes careless mistakes in schoolwork, chores, or other activities?" It characterizes one side of a child's behavior for recognizing inattention, but to know the overall picture of the child's behavior the clinician needs to know the combination of the answers on all the nine questions.

According to these questions, we define nine criteria $G = \{g_1, \dots, g_9\}$ with equal importance for the clinician, two values on each criterion $g_j = \{g_{j1}, g_{j2}\} = \{rarely,\ often\}$, $(j = 1, \dots, n)$ and two classes $L = \{l_1,\ l_2\} = \{no\ inattention,\ inattention\ \ exists\}$ that are ordered in an increasing order. Here, we are solving a maximization problem, where the maximal ranks of criteria values and maximal rank of class correspond to strongest affirmation of inattention. Then, the Cartesian product of the scales of criteria allows us to obtain the set of alternatives X with $|X| = 2^9 = 512$, that is, we have 512 combinations of answers on each of the nine questions and $X = \{x_1, \ldots, x_{512}\}$, where

$$\begin{aligned}
&x_1 = (rarely, rarely, \ldots, rarely),\\
&x_2 = (often, rarely, \ldots, rarely),\\
&\ldots\\
&x_{511} = (often, often, \ldots, often, rarely),\\
&x_{512} = (often, often, \ldots, often, often).
\end{aligned} \tag{24}$$

Thus, it is necessary to classify the five hundred twelve alternatives into two classes.

The application of the Dichotomic Classification method to the diagnostics of ADHD

Step 1 The algorithm starts by assigning the set of possible classes $L_i^{old} = L_i = L = \{l_1, l_2\}$ for each alternative and allocating the alternative with all the maximal ranks (largest criteria values), i.e., $x_{512} = (often, often, \ldots, often, often)$, into the class with the maximal rank $x_{512} \in l_2$, where $l_2 = inattention\ exists$, and alternative with all the minimal ranks (smallest criteria values), i.e., $x_1 = (rarely, rarely, \ldots, rarely)$, into the class with the minimal rank $x_1 \in l_1$, where $l_1 = no\ \ inattention$.

Iteration 1

Step 2 Here, we should check whether there are not yet classified alternatives for which $|L_i| = 1$. There are 510 not yet classified alternatives $X_L = \{x_2, \ldots, x_{511}\}$.

Step 3 The set of not yet classified alternatives is bigger than two; thus, we should search for the middle alternative with regards to the alternative x_1 with the minimal ranks on all criteria and the alternative x_{512} with the maximal ranks on all criteria as proposed in *Step 3* of Section 4.2. However, in the problem discussed there are only two values on the scale of each criterion, and there is no sense in searching for the middle values on the scale of each criterion. This fact simplifies the procedure of selecting a middle alternative: we may choose it according to the bisection on the set of not yet classified alternatives (as in the case with ordered criteria). In this way, we obtain the first middle alternative $x_{255} = (rarely, often, often, often, often, often, often, often, rarely)$.

Step 4 The middle alternative is proposed to the DM for classification. According to the rule that imitates a clinician (for the conclusion about inattention diagnosis at least six symptoms of inattention should appear), the alternative x_{255} is assigned to the class $l_2 = inattention\ exists$ (seven symptoms appear).

Step 5 There is no need to check inconsistencies because we have only two classes (see Section 4.2).

Step 6 At this step, the set of possible classes for the not yet classified alternatives is recalculated according to the classification of the alternative at *Step 4*. We check the preference relation for the alternative classified by the DM at *Step 4* and each not yet classified alternative. We find that the alternatives $x_{256} = (rarely, often, often, often, often, often, often, often, often)$ and $x_{511} = (often, often, often, often, often, often, often, often, rarely)$ are preferred to the alternative $x_{256} = (rarely, often, often, often, often, often, often, often, rarely)$: $x_i P x_{255}$,

$i = 256, 511$. According to the relation (12), the set of possible classes for the alternatives x_{256} and x_{511} should be recalculated. For instance, for the alternative x_{256}, we have $x_{256}Px_{255}$, and, thus, $L_{256} = L_{256}^{old} \bigcap l_2 = l_1, l_2 \bigcap l_2 = l_2$ (*inattention exists*). In such a way, the alternatives x_{256} and x_{511} are classified indirectly: $x_{256} \in l_2$ and $x_{511} \in l_2$. After this step, the next iteration is performed in a similar way starting from *Step 2*.

Iteration 2

Step 2 We should check whether there are not yet classified alternatives. At this moment, the set of not yet classified alternatives is $X_L = \{x_2, \ldots, x_{254}, x_{257}, \ldots, x_{510}\}$.

Step 3 We continue the search for the middle alternatives according to the bisection on both parts of the set of not yet classified alternatives. At this step, we find two middle alternatives x_{128} and x_{383}.

Step 4 The selected alternatives are proposed to the DM for classification one by one at the current and next iterations, respectively. According to the assumed rule (inattention is concluded when six or more symptoms appear), the alternative $x_{128} = (rarely, rarely, often, often, often, often, often, often, often)$ is assigned to the class $l_2 = inattention\ \ exists$.

Step 5 There is no need to check inconsistencies because we have only two classes (see Section 4.2).

Step 6 Assignment of the alternative x_{128} into the class l_2 allows indirect classification of the alternative $x_{384} = (often, rarely, often, often, often, often, often, often, often)$ into the class $l_2 = inattention\ exists$ with regards to the preference relation $x_{384}Px_{128}$: $L_{384} = L_{384}^{old} \bigcap l_2 = \{l_1, l_2\} \bigcap l_2 = l_2$. The iterative procedure continues until all five hundred twelve alternatives are classified.

Thus, with the Dichotomic Classification method, 242 questions are proposed to the clinician (when compared to 336 questions that should be asked by the SAC method). On the other hand, the Dichotomic Classification method needs no more than 250 milliseconds of CPU time when selecting an alternative to pose to the DM for classification at each iteration. SAC, on the other hand, requires 9000 milliseconds of CPU time at first iteration, and that time is linearly decreasing for the following iteration. This indicates that the computational complexity when selecting an alternative at each iteration is considerably decreased and that the effectiveness (in the number of questions posed to the DM) is improved in Dichotomic Classification when compared to SAC.

Eventually, after the classification process, the alternatives are assigned in the following way: from the whole set of 512 possible combinations of the symptoms, 382 do not indicate inattention and 130 define this disorder. The resulting table is partially presented in Figure 34. Due to the similarity of the classification rule for the detection of hyperactivity-impulsivity, the number of questions posed to the DM is the same as for the indication of inattention: 242 when the Dichotomic Classification method is applied and 336 when the SAC method is used. At the third stage of the diagnostics, we aggregate the results of the two previous stages and obtain the evaluation for the ADHD: 33 alternatives from

512 meet both subtypes of criteria (inattention and hyperactivity-impulsivity), and 285 alternatives do not have any of the ADHD subtypes.

7.2.3 Concluding Remarks

We have discussed ADHD diagnostics with the help of the interactive Dichotomic Classification method for ordinal classification. The tests were done in the framework of a multicriteria decision support system presented in Section 7.3 in cooperation with the Niilo Mäki Institute (Jyväskylä, Finland), where there is considerable experience in solving such problems. The system can assist an expert as well as a new specialist in neuropsychological diagnostics.

In order to demonstrate the method, we assumed some rules that imitate evaluations made by clinician. However, there is a more realistic situation with some uncertainty on qualitative estimations. We can define such condition with "unsure" value on the scale of each criterion: $g_j = \{g_{j1}, g_{j2}, g_{j3}\} = \{rarely, unsure, often\}$. In this case, the aid of multicriteria decision support system is indispensable, since it allows defining the rules that are "hidden" even from the clinician. This case is the subject for future research. There is another source for our investigation: the use of additional information, such as distribution of the alternatives between classes (in %, for instance) and utilization of data bases of the known diagnostic cases that can be used as assignment examples. Next, we have a look at the multicriteria decision support system that contains implementations of the SAC method studied and the SMAA-Classification and Dichotomic Classification methods proposed in this work.

7.3 Multicriteria Decision Support System for Classification

The multicriteria decision support system (MDSS) is a powerful tool that is aimed to assist the analyst and/or DM when solving multicriteria problems. Indeed, such system may help at different stages of the decision aiding process that include (but not limited to) structuring main components of the model and formulating a task, defining preferences of the DM and selecting a method to be used.

When developing an MDSS for classification, we take into account the fact that the DM is not necessarily familiar with the MCDA methodology. That is why the system developed should be interactive and iterative, and it should serve the learning of the DM through a dialog with the system. The interface should be user-friendly and should provide necessary graphical or visualization tools. The wishes of the DM should be considered and psychological as well as behavioral aspects should be taken into account. The system should be platform independent and should preferably be available via Internet. In this work, we have developed MDSS taking into account the desired features.

Let us briefly discuss the MDSS developed. The system allows classification with one of the methods, SMAA-Classification, SAC or Dichotomic Classification methods. We have developed MDSS with Eclipse 3.1.1 Java Software Development Kit using a Celeron (R) Processor equipped PC with 2 GHz and RAM 256. It is platform independent (can be used at, e.g., Windows, Linux, Unix). We demonstrate some screenings of the MDSS when solving the ADHD diagnostics problem presented in Section 7.2.2.

Figure 32 shows the initial information input in the form of criteria with name, description and a scale that can have verbal or numerical values for each criterion and classes with a name and description for each class (and assignment examples for SMAA-Classification). The user (analyst and/or DM) has possibilities to add, to remove, to update and to save initial, resulting and intermediate information. The system creates the set of alternatives (called objects at the screenings at Figures 32-34) as a Cartesian product of criteria scales and allows deselecting the alternatives that are not currently given (even though they are potential) in the problem. We are also planning to add the possibility of a more natural but time-consuming (especially, for the large data sets) way to specify the alternatives: the input of each given alternative by the DM.

After the initial information input, the classification method should be selected. Currently, it is possible in the MDSS to assign alternatives estimated on criteria into classes by means of SMAA-Classification methods, SAC or Dichotomic Classification. For SMAA-Classification, all alternatives are classified automatically, but the set of assignment examples has to be predefined. For the interactive VDA methods (SAC and Dichotomic Classification), there is possibility to specify lexicographic order of criteria. When the interactive VDA method is selected, the DM is asked to classify some alternatives from the set of given alternatives (but not all). The alternatives to be classified by the DM are posed to the screen one by one, and he or she should select one class from the set of predefined classes for each alternative. Figure 33 shows one of the iterations of such a questioning procedure.

When the method has run its course, the results of classification can be browsed; an example of such screening is presented in Figure 34. Except for standard functions for the creation and modification of the problem by adding, removing and editing the criteria and classes sets, there are options for saving and opening separate files with a problem model or with the DM interview or with the classification results.

By using an MDSS for classification, the analyst and/or the DM can get a significant help at different stages of the decision aiding process. At the same time, the analyst and/or the DM can learn more about the problem considered and the methodology applied.

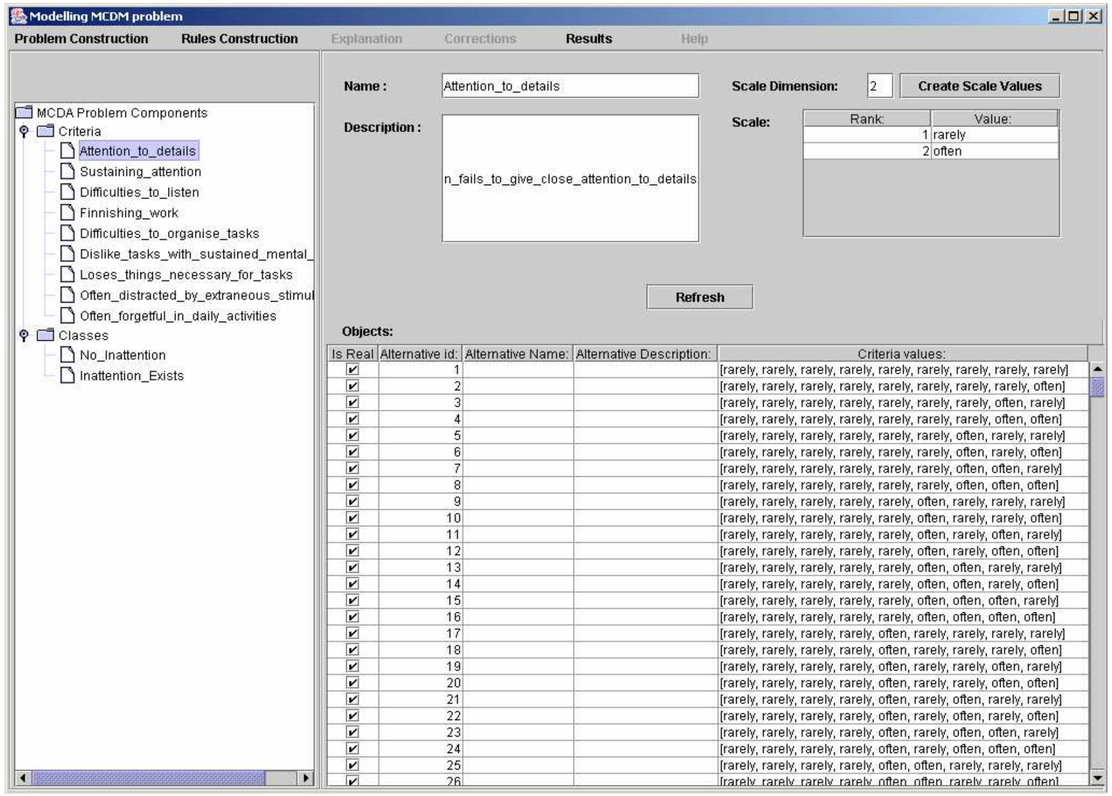

FIGURE 32 Screen of the initial information input

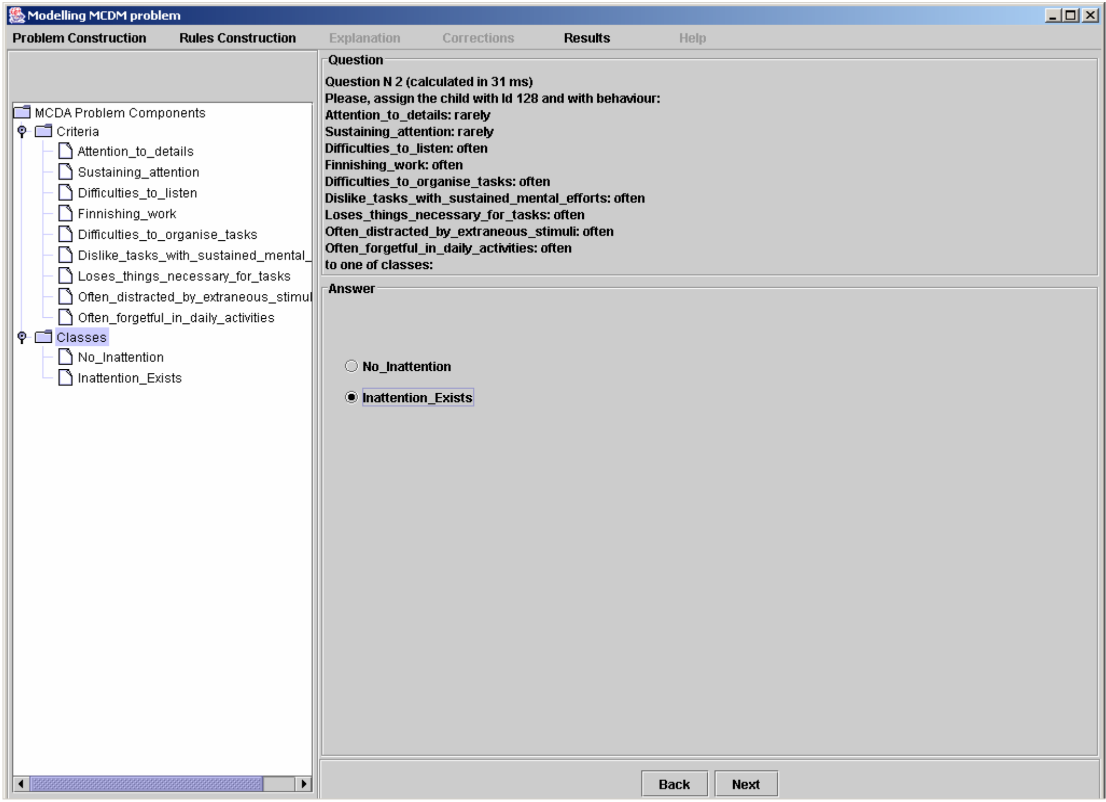

FIGURE 33 Screen of the questioning procedure iteration

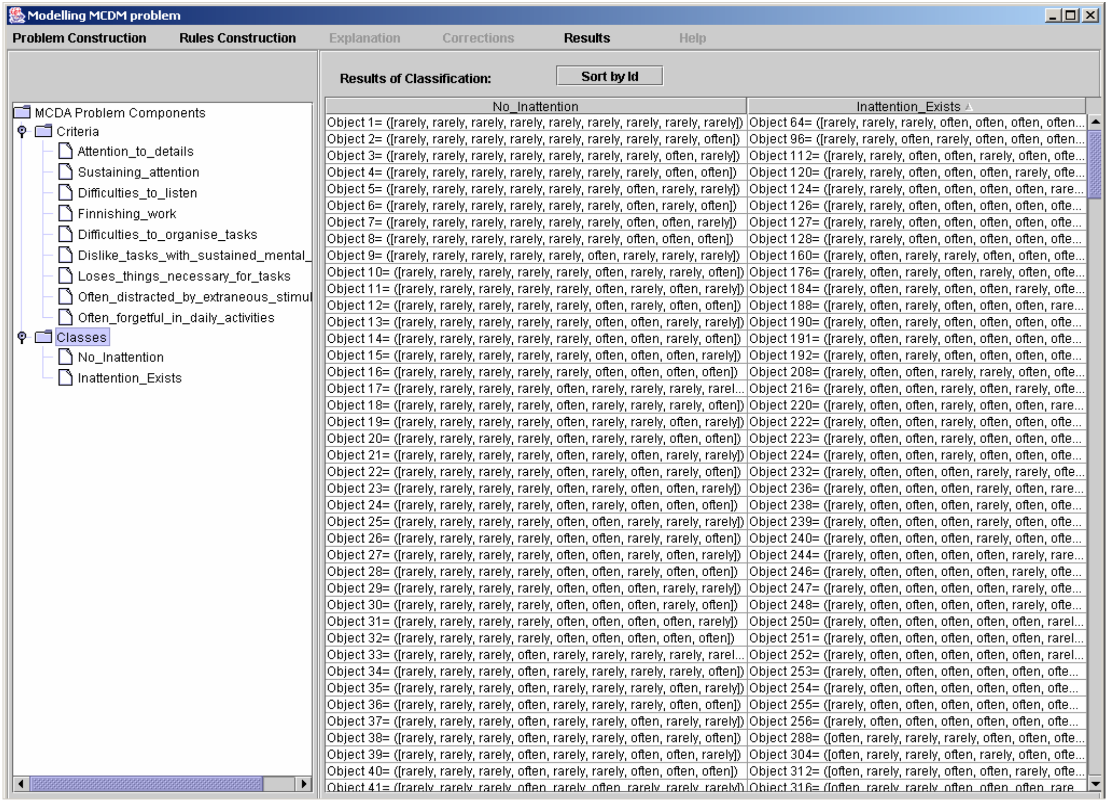

FIGURE 34 Screen of the classification results

8 CONCLUSIONS AND FUTURE RESEARCH

The main motivation for this work was to structure the field of the MCDA-based classification. We have discussed classification problems and the ways they are solved by MCDA approaches. At the beginning, we have introduced some basic concepts and theories of multicriteria decision aiding. Then, we have defined the classification problem and presented a comprehensive survey of the MCDA methods existing in the literature. We have described the most widely-spread MCDA methods for classification in more detail.

According to the assumption about the possibility to describe the behavior of the DM by means of some model, we have categorized the MCDA methods for classification on parameter-based and parameter-free methods. In turn, parameter-based methods (that can be utility function-based and outranking relation-based) are divided into direct and indirect ones according to the ability of the DM to provide parameters of the selected method. Thus, the methods have been categorized into three main groups: direct parameter-based, indirect parameter-based and parameter-free. Most of the methods presented in this work are from the last two groups.

The comparative analysis of the methods surveyed shows that there are some gaps in existing MCDA methods and it is possible to improve some of the existing methods as well as extend some MCDA methodology for the classification problems. Thus, we have developed two new methods: SMAA-Classification and Dichotomic Classification in the framework of indirect parameter-based and parameter-free groups of methods, respectively. The methods developed differ not only according to the group they belong to but also with regards to the initial data used, resulting data provided to the DM, and the type of the classification problem they solve.

The nominal SMAA-Classification method has been developed for the cases, where it is assumed that the behavior of the DM can be described by some parameter-based model; however, the DM has difficulties to define the precise parameter values of the model. The method does not need them to be defined by the DM; nevertheless, if there is some imprecise or uncertain information about them available, it can be used by the method. On the other hand, the SMAA-Classification method assumes that the DM can specify some assignment examples for each class. Such assignment examples may be decisions that the DM used to make in the past. Alternatively, he or she may be willing to think about the possible consequences of decisions made for realistic alternatives not considered in the set of given alternatives, or can make decisions for a subset of alternatives from the set of given alternatives. The SMAA-Classification method allows nominal classification of given sets of alternatives described by numerical criteria or attributes. The method is based on the Monte-Carlo simulation. At each iteration, the possible preferences of the DM are modeled. Taking them into account, the classification of the set of given alternatives is done based on the minimal distance between the alternative to be classified and assignment examples of classes. After running the specified number of simulations, statistical information about frequency of each alternative to be assigned to each class as well as typical preferences that support such an assignment is recorded.

As far as the Dichotomic Classification method is considered, it has been developed in the framework of VDA and it inherits the main features of this approach. Indeed, Dichotomic Classification is an interactive method for ordinal classification that asks the DM to classify some, but not all, the alternatives from the set of given alternatives. The rest of the alternatives are classified indirectly. The selection of the alternative to be posed to the DM for direct classification is done based on the dichotomic search on the set of alternatives. The Dichotomic Classification method does not require transformation of the verbal values on the criteria scales to numerical analogies. The alternatives estimated on verbal values are posed to the DM in a natural language. Questioning of the DM continues until each alternative has been assigned to exactly one class. It is assumed that the preferences of the DM may change over time and that during the interrogation procedure the DM may be inconsistent. That is why the classification at each iteration is compared to the previously made ones. In case any contradiction appears, the DM is asked to rethink both or one of the contradicting classifications.

We have also considered the possibility of additional information to be taken into account in VDA methods, such as lexicographic order on the set of criteria. This information, if available, simplifies the classification procedure and reduces the number of questions posed to the DM for direct classification.

Even though both the new methods introduced here have been developed in the framework of MCDA theory, they aim at different types of classification problems. Thus, they have different assumptions, different initial and resulting data and a different process of decision aiding itself. We have described the functioning of the methods with the help of simple illustrative examples. We have also tested and compared their performance to the methods developed earlier on data sets available in the MCDA literature. The analysis of the properties of methods studied and developed allows us to suggest some recommendations for selecting a method to be used depending on the initial information, requirements to the resulting data, type of classification, and the decision aiding process. Thus, we have formulated decision trees to support the task of selecting a method for classification.

We have also created a multicriteria decision support system in the framework of which the developed and some of the methods studied have been implemented and tested. Besides the methodological part, this software contains a user-friendly interface and allows a comfortable interaction between the user (the analyst and/or DM) and the methods.

Finally, the Dichotomic Classification method has been applied for the diagnostics of Attention-Deficit Hyperactivity Disorder. These tests have been done in cooperation with neuropsychologists at the Niilo Mäki Institute (Jyväskylä, Finland). In the future, it would be interesting to apply the methods developed to other diagnostic problems.

To summarize, the main contributions of this work are following. We have been motivated by the idea of structuring the field of classification with MCDA approaches. For this purpose, we have presented a comprehensive state-of-the-art of classification with MCDA approaches. We have categorized the presented approaches, compared them and provided recommendations for selecting a method in a particular situation. This survey allows us to realize gaps existing in the field of MCDA classification. In order to fill some gaps, we have developed two new methods. We have implemented the methods developed in the framework of the multicriteria decision support system and have applied one of the methods developed to the neuropsychological diagnostics. At the same time, some other gaps remain in this area and there is still a lot of work to be done in the future. Next, we point out some directions of research that are worth of close attention.

Even though research in the area of classification with MCDA approaches has evolved rapidly over the past two decades, there are still unsolved problems left and interesting topics that need further investigation. Indeed, most of the research done so far has concentrated on the development of new methods. Even though some gaps still exist and there is space for new methods to fill them, studying the methods developed should continue, for example, in the following directions.

The effectiveness and complexity of the existing methods should be estimated for different data sets. Currently, with the exception of work done in [37], there are not many comparisons and applications of different MCDA classification methods to the same data sets documented in the MCDA literature. The research in this direction should be extended also because real-life applications and practical situations may improve existing method and create challenges for the development of new ones. That is why a repository with benchmark real-life applications and artificial data sets should be created for testing the MCDA classification methods. Other important issues to be discussed are estimation of the validity and quality of the methods developed as well as sensitivity and robustness analysis.

In a similar way presented in this work, the comparison of different methods and recommendations for selecting a method to be used in a particular situation can be investigated further. Another important issue to be studied is using combinations of results of several MCDA classification methods, for instance, the way as it is done in machine learning with voting procedures [39].

Recently, a large number of classification methods have been developed within the field of artificial intelligence, in particular, in machine learning. The effectiveness of the MCDA methods can be compared to non-parametric techniques, for instance, to neural networks, expert systems, and decision trees. In a similar way as it is done in [37], the MCDA methods can be compared to the statistical analysis tools such as discriminant analysis. On the other hand, the methods can be combined or hybridized in such a way that each method works only at a certain stage of the decision aiding process.

Similarly to the combination of the ELECTRE TRI method and the SMAA methodology in the SMAA-TRI method, the SMAA-Classification method can be applied at the initial stages of the VDA methods. We plan to develop a VDA-SMAA method that would perform classification of the given sets of alternatives described with verbal or numerical criteria scales to ordered classes, the number of which would be predefined, but the boundary alternatives are unknown a priori. (The assignment examples are assumed to be available in advance.) At the beginning of the VDA-SMAA method, SMAA-Classification should perform pre-classification and define acceptabilities of each alternative to be assigned to each class. Then, the alternatives that have large acceptabilities to be assigned to several classes should be posed to the DM for direct classification.

In the future, it would also be interesting to compare the application of the methods of more general types of classification problems, for instance, clustering and nominal classification, with methods of more specific types of classification problems, for instance, nominal classification and ordinal classification, to the same data sets. In this way, we would be able to check whether methods developed for more general and more specified cases of classification provide similar results on the same data sets, and whether it is possible to detect the specific features after application of the more general methods. For instance, it would be interesting to test if it is possible to discover the order of the classes after application of the nominal classification method to the data set with no information about the order between classes available a priori.

Finally, we wish to emphasize the need for multicriteria decision support systems that would be able to provide efficient assistance to the DM when selecting and applying the MCDA methods for classification problems. They should have user-friendly interfaces and should provide the DM with necessary graphical or visualization tools. Such systems should be developed taking into account the requirements and wishes of real DMs.

REFERENCES

[1] B. Ahn, "Extending Malakooti's model for ranking multicriteria alternatives with preference strength and partial information," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 33, no. 3, pp. 498–507, 2003.

[2] American Psychiatric Association, *Diagnostic and statistical manual of mental disorders*, 4th ed. Washington: DC, 1994.

[3] C. Arondel and P. Girardin, "Sorting cropping systems on the basis of their impact on groundwater quality," *European Journal of Operational Research*, vol. 127, no. 3, pp. 467–482, 2000.

[4] Association of European Operational Research Societies. [Online]. Available: http://www.euro-online.org/

[5] R. Azibi and D. Vanderpooten, "Construction of rule-based assignment models," *European Journal of Operational Research*, vol. 138, no. 2, pp. 274–293, 2002.

[6] C. Bana e Costa, "A multicriteria decision aid methodology to deal with conflicting situations on the weights," *European Journal of Operational Research*, vol. 26, no. 1, pp. 22–34, 1986.

[7] C. Bana e Costa and R. Oliveira, "Assigning priorities for maintenance, repair and refurbishment in managing a municipal housing stock," *Journal of Operational Research*, vol. 138, pp. 380–391, 2002.

[8] A. Bastian, "Identifying fuzzy models utilizing genetic programming," *Fuzzy Sets and Systems*, vol. 113, no. 3, pp. 333–350, 2000.

[9] N. Belacel, "Méthodes de classification multicritère: Méthodologie et application au diagnostic médical," Thèse de Doctorat, Université Libre de Bruxelles (ULB), Bruxelles, Belgique, 1998.

[10] ——, "Multicriteria assignment method PROAFTN: Methodology and medical application," *European Journal of Operational Research*, vol. 125, no. 1, pp. 175–183, 2000.

[11] N. Belacel and M. Boulassel, "Multicriteria fuzzy assignment method: A useful tool to assist medical diagnosis," *Artificial Intelligence in Medicine*, no. 21, pp. 201–207, 2001.

[12] ——, "Multicriteria fuzzy classification procedure PROCFTN: Methodology and medical application," *Fuzzy Sets and Systems*, vol. 141, no. 2, pp. 203–217, 2004.

[13] N. Belacel, P. Vincke, J. Scheiff, and M. Boulassel, "Acute leukemia diagnosis aid using multicriteria fuzzy assignment methodology," *Computer Methods and Programs in Biomedicine*, no. 64, pp. 145–151, 2001.

[14] N. Belacel, H. B. Raval, and A. P. Punnen, "Learning multicriteria fuzzy classification method PROAFTN from data," *Computers & Operations Research*, vol. 34, no. 7, pp. 1885–1898, 2007.

[15] R. Bellman and L. Zadeh, "Decision-Making in a Fuzzy Environment," *Management Science*, vol. 17, 1970.

[16] V. Belton and T. Stewart, *Multiple Criteria Decision Analysis: An Integrated Approach*. Dordrecht: Kluwer Academic Publishers, 2002.

[17] J. Blaszczynski, S. Greco, and R. Slowinski, "Multi-criteria classification - a new scheme for application of dominance-based decision rules," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1030–1044, 2007.

[18] C. I. Bliss, "The method of probits," *Science*, vol. 79, pp. 38–39, 1934.

[19] D. Bouyssou, T. Marchant, M. Pirlot, P. Perny, A. Tsoukiàs, and P. Vincke, *Evaluation and decision models: A critical perspective*. Dordrecht: Kluwer Academic Publishers, 2000.

[20] D. Bouyssou and T. Marchant, "An axiomatic approach to noncompensatory sorting methods in MCDM, I: The case of two categories," *European Journal of Operational Research*, vol. 178, no. 1, pp. 217–245, 2007.

[21] ——, "An axiomatic approach to noncompensatory sorting methods in MCDM, II: More than two categories," *European Journal of Operational Research*, vol. 178, no. 1, pp. 246–276, 2007.

[22] D. Bouyssou, T. Marchant, M. Pirlot, A. Tsoukias, and P. Vincke, *Evaluation and decision models with multiple criteria: Stepping stones for the analyst*, 1st ed. Boston: Springer-Verlag, 2006.

[23] J. Brans and B. Mareschal, "PROMETHEE methods," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston: Springer-Verlag, 2005, pp. 163–196.

[24] J. Charnetski and R. Soland, "Multiple-attribute decision making with partial information: The comparative hypervolume criterion," *Naval Research Logistics Quarterly*, vol. 25, pp. 279–288, 1978.

[25] Y. Chen, D. Kilgour, and K. Hipel, "Multiple criteria classification with an application in water resources planning," *Computers & Operations Research*, vol. 33, no. 11, pp. 3301–3323, 2006.

[26] Y. Chen, "Multiple criteria decision analysis: Classification problems and solutions," Ph.D. dissertation, University of Waterloo, Waterloo, Ontario, Canada, 2006.

[27] O. Cordon, M. del Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification systems," *International Journal of Approximate Reasoning*, vol. 20, pp. 21 – 45, 1999.

[28] D. Crystal, R. Ostrander, R. Chen, and G. August, "Multimethod assessment of psychopathology among DSM-IV subtypes of children with Attention-Deficit / Hyperactivity Disorder: Self-, parent, and teacher reports," *Journal of Abnormal Child Psychology*, vol. 29, no. 3, pp. 189–205, 2000.

[29] J. Devaud, G. Groussaud, and E. Jacquet-Lagreze, "UTADIS: Une methode de construction de fonctions d'utilite additives rendant compte de jugements globaux," in *Proceedings of European working group on MCDA*, Bochum, Germany, 1980.

[30] L. Dias and V. Mousseau, "IRIS: A DSS for multiple criteria sorting problems," *Journal of Multi-Criteria Decision Analysis*, vol. 12, pp. 285–298, 2003.

[31] ——, "Inferring ELECTRE's veto-related parameters from outranking examples," *European Journal of Operational Research*, vol. 170, no. 1, pp. 172–191, 2006.

[32] L. Dias, V. Mousseau, J. Figueira, and J. Clímaco, "An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI," *European Journal of Operational Research*, vol. 138, no. 2, pp. 332–348, 2002.

[33] L. Dias and A. Tsoukiàs, "On the constructive and other approaches in decision aiding," in *Proceedings of the 56th meeting of the EURO MCDA working group*, C. Hengeller Antunes, J. Figueira, and J. Clímaco, Eds. Coimbra: CCDRC, 2004, pp. 13–28.

[34] A. Dimitras, C. Zopounidis, and C. Hurson, "A multicriteria decision aid method for the assessment of business failure risk," *Foundations of Computing and Decision Sciences*, vol. 20, no. 2, pp. 99–112, 1995.

[35] M. Doumpos, K. Kosmidou, G. Baourakis, and C. Zopounidis, "Credit risk assessment using a multicriteria hierarchical discrimination approach: A comparative analysis," *European Journal of Operational Research*, vol. 138, no. 2, pp. 392–412, 2002.

[36] M. Doumpos and C. Zopounidis, "The use of the preference disaggregation analysis in the assessment of financial risks," *Fuzzy Economics Review*, vol. 31, pp. 39–57, 1998.

[37] ——, *Multicriteria Decision Aid Classification Methods*. Dordrecht: Kluwer Academic Publishers, 2002.

[38] ——, "A multicriteria classification approach based on pairwise comparisons," *European Journal of Operational Research*, vol. 158, no. 2, pp. 378–389, 2004.

[39] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: J. Wiley, 2000.

[40] M. Ehrgott, *Multicriteria Optimization*. Berlin: Springer-Verlag, 2000.

[41] M. Ehrgott and X. Gandibleux, *Multiple Criteria Optimization. State of the art annotated bibliographic surveys*. Dordrecht: Kluwer Academic Publishers, 2002.

[42] H. Eiselt and G. Laporte, "The use of domains in multicriteria decision-making," *European Journal of Operational Research*, vol. 61, no. 3, pp. 292–298, 1992.

[43] EURO Working Group on MultiCriteria Decision Aiding. [Online]. Available: http://www.inescc.pt/ ewgmcda/

[44] J. Figueira, S. Greco, and M. Ehrgott, Eds., *Multiple Criteria Decision Analysis: State of the Art Surveys*. Boston: Springer-Verlag, 2005.

[45] J. Figueira, V. Mousseau, and B. Roy, "Electre methods," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston: Springer-Verlag, 2005, pp. 133–162.

[46] J. Figueira and B. Roy, "Determining the weights of criteria in the ELECTRE type methods with a revised Simos' procedure," *European Journal of Operational Research*, vol. 139, pp. 317–326, 2002.

[47] J. Figueira, Y. Smet, and J. Brans, "MCDA methods for sorting and clustering problems: Promethee TRI and Promethee CLUSTER," Universite' Libre de Bruxelles, SMG," Technical Report TR/SMG/2004-002, 2004.

[48] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[49] N. Flanders, R. Brown, Y. Andre'eva, and O. Larichev, "Justifying public decisions in arctic oil and gas development: American and Russian approaches," *Arctic*, vol. 3, no. 51, pp. 262–279, 1998.

[50] E. Furems and L. Gnedenko, "STEPCLASS - a system for eliciting expert knowledge and conducting expert analysis for solving diagnostic problems," *Automatic Documentation and Mathematical Linguistics Translations of Selected Articles from Nauchno-Tekhnicheskaia Informasiia*, vol. 30, no. 5, pp. 23–28, 1996.

[51] E. Furems, O. Larichev, G. Roizenson, A. Lotov, and K. Miettinen, "Human behavior in a multi-criteria choice problem with individual tasks of different difficulties," *International Journal of Information Technology & Decision Making*, vol. 2, no. 1, pp. 29–40, 2003.

[52] T. Gal, T. Hanne, and T. Stewart, Eds., *Advances in Multiple Criteria Decision Making*. Dordrecht: Kluwer Academic Publishers, 1998.

[53] E. Georgopoulou, Y. Sarafidis, S. Mirasgedis, S. Zaimi, and D. Lalas, "A multiple criteria decision-aid approach in defining national priorities for greenhouse gases emissions reduction in the energy sector," *European Journal of Operational Research*, vol. 146, no. 1, pp. 199–215, 2003.

[54] C. Gillberg, *Clinical Child Neuropsychiatry*. Cambridge: Cambridge University Press, 1995.

[55] S. Greco, B. Matarazzo, and R. Slowinski, "A new rough set approach to evaluation of bankruptcy risk," in *Operational Tools in the Management of Financial Risks*, C. Zopounidis, Ed. Dodrecht: Kluwer Academic Publishers, 1998, pp. 121–136.

[56] S. Greco, B. Matarazzo, and R. Slowinski, "The use of rough sets and fuzzy sets in MCDM," in *Advances in MCDM Models, Algorithms, Theory, and Applications*, T. Gal, T. Stewart, and T. Hanne, Eds. Dordrecht: Kluwer Academic Publishers, 1999, pp. 14.1–14.59.

[57] S. Greco, B. Matarazzo, and R. Slowinski, "Rough sets theory for multicriteria decision analysis," *European Journal of Operational Research*, vol. 129, no. 1, pp. 1–47, 2001.

[58] ——, "Rough approximation by dominance relations," *International Journal of Intelligent Systems*, vol. 17, no. 2, pp. 153–171, 2002.

[59] ——, "Rough sets methodology for sorting problems in presence of multiple attributes and criteria," *European Journal of Operational Research*, vol. 138, pp. 247–259, 2002.

[60] ——, "Decision rule approach," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston: Springer-Verlag, 2005, pp. 507–562.

[61] S. Greco, B. Matarazzo, R. Slowinski, and J. Stefanowski, "Variable consistency model of dominance-based rough set approach," in *Rough Sets and Current Trends in Computing*, W. Ziarko and Y. Yao, Eds. Berlin: Springer-Verlag, 2001, pp. 170–181.

[62] S. Greco and at al., "Sessions dominance-based rough set approach I-III," in *Book of Abstracts EURO XXII Prague July 8-11*, J. Jablonsky, Ed., 2007.

[63] J. Grzymala-Busse, "LERS - A system for learning from examples based on rough sets," in *Intelligent Decision Support - Handbook of Applications and Advances of the Rough Set Theory*, R. Slowinski, Ed. Dordrecht: Kluwer Academic Publishers, 1992, pp. 3–18.

[64] A. J. Hartigan, *Clustering Algorithms*. New York: J. Wiley, 1975.

[65] J. Hokkanen, R. Lahdelma, and P. Salminen, "A multiple criteria decision model for analyzing and choosing among different development patterns for the Helsinki cargo harbor," *Socio-Economic Planning Sciences*, vol. 33, pp. 1–23, 1999.

[66] N. Hovanov, M. Yudaeva, and K. Hovanov, "Multicriteria estimation of probabilities on basis of expert non-numeric, non-exact and non-complete knowledge," *European Journal of Operational Research*, 2007, (to appear). [Online]. Available: doi:10.1016/j.ejor.2007.11.018

[67] International Society on Multiple Criteria Decision Making. [Online]. Available: http://www.terry.uga.edu/mcdm/index.html

[68] M. Inuiguchi, T. Tanino, and M. Sakawa, "Membership function elicitation in possibilistic programming problems," *Fuzzy Sets and Systems*, vol. 111, no. 1, pp. 29–45, 2000.

[69] H. Ishibuchi, K. Nozaki, and H. Tanaka, "Distributed representation of fuzzy rules and its application to pattern classification," *Fuzzy Sets and Systems*, vol. 52, no. 1, pp. 21–32, 1992.

[70] ——, "Efficient fuzzy partition of pattern space for classification problem," *Fuzzy Sets and Systems*, vol. 59, no. 3, pp. 295–304, 1993.

[71] E. Jacquet-Lagrèze and Y. Siskos, "Assessing a set of additive utility functions for multicriteria decision making: The UTA method," *European Journal of Operational Research*, vol. 10, pp. 151–164, 1982.

[72] ——, "Preference disaggregation: 20 years of MCDA experience," *European Journal of Operational Research*, vol. 130, no. 2, pp. 233–245, 2001.

[73] A. Jain, M. Murty, and P. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[74] A. Jaszkiewicz and A. B. Ferhat, "Solving multiple criteria choice problems by interactive trichotomy segmentation," *European Journal of Operational Research*, vol. 113, no. 2, pp. 271–280, 1999.

[75] F. Joerin and A. Musy, "Land management with GIS and multicriteria analysis," *International Transactions in Operational Research*, vol. 7, no. 1, pp. 67–78, 2000.

[76] D. Kahneman and A. Tversky, "Prospect theory: An analysis of decision under risk," *Econometrica*, vol. 47, pp. 263–291, 1979.

[77] A. Kalff, J. Hendriksen, M. Kroes, J. Vles, J. Steyaert, F. Feron, T.M.C.B. van Zeben, and J. Jolles, "Neurocognitive performance of 5- and 6-year-old children who met criteria for Attention Deficit / Hyperactivity Disorder at 18 months follow-up: Results from a prospective population study," *Journal of Abnormal Child Psychology*, vol. 30, no. 6, pp. 589–598, 2002.

[78] J. Kangas, J. Hokkanen, A. Kangas, R. Lahdelma, and P. Salminen, “Applying stochastic multicriteria acceptability analysis to forest ecosystem management with both cardinal and ordinal criteria,” *Forest Science*, vol. 49, no. 6, pp. 928–937, 2003.

[79] J. Kangas and A. Kangas, “Multicriteria approval and SMAA-O method in natural resources decision analysis with both ordinal and cardinal criteria,” *Journal of Multi-Criteria Decision Analysis*, vol. 12, pp. 3–15, 2003.

[80] R. Keeney and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. New York: J. Wiley, 1976.

[81] D. Kochin, “Classification method of given objects set,” Master’s thesis, Moscow Institute for Physics and Technology, Moscow, Russia, 2002, (In Russian).

[82] M. Koksalan and C. Ulu, “An interactive approach for placing alternatives in preference classes,” *European Journal of Operational Research*, vol. 144, pp. 429–439, 2003.

[83] R. Lahdelma, J. Hokkanen, and P. Salminen, “SMAA - Stochastic Multicriteria Acceptability Analysis,” *European Journal of Operational Research*, vol. 106, no. 1, pp. 137–143, 1998.

[84] R. Lahdelma, S. Makkonen, and P. Salminen, “Treating dependent uncertainties in multi-criteria decision problems,” in *Operational Research and Its Applications: Recent Trends*, M. Rao and M. Puri, Eds. New Delhi: Allied Publishers Pvt. Ltd, 2004, vol. 2, pp. 427–434.

[85] R. Lahdelma, K. Miettinen, and P. Salminen, “Ordinal criteria in stochastic multicriteria acceptability analysis (SMAA),” *European Journal of Operational Research*, vol. 147, no. 1, pp. 117–127, 2003.

[86] ——, “Reference point approach for multiple decision makers,” *European Journal of Operational Research*, vol. 164, no. 3, pp. 785–791, 2005.

[87] R. Lahdelma and P. Salminen, “SMAA-2: Stochastic multicriteria acceptability analysis for group decision making,” *Operations Research*, vol. 49, no. 3, pp. 444–454, 2001.

[88] ——, “Pseudo-criteria versus linear utility function in stochastic multicriteria acceptability analysis,” *European Journal of Operational Research*, vol. 141, no. 2, pp. 454–469, 2002.

[89] ——, “Classifying efficient alternatives in SMAA using cross confidence factors,” *European Journal of Operational Research*, vol. 170, no. 1, pp. 228–240, 2006.

[90] ——, “Loss aversion in SMAA,” University of Jyväskylä, School of Business and Economics,” Tech. Rep. 269, 2003.

[91] LAMSADE laboratory of University of Paris-Dauphine, Paris, France. [Online]. Available: http://www.lamsade.dauphine.fr/mcda/biblio/

[92] O. Larichev, "A study on the internal organization of expert knowledge," *Pattern Recognition and Image Analysis*, vol. 5, no. 1, pp. 57–63, 1995.

[93] O. Larichev, A. Asanov, and Y. Naryzhny, "Effectiveness evaluation of expert classification methods," *European Journal of Operational Research*, vol. 138, no. 2, pp. 260–273, 2002.

[94] O. Larichev and R. Brown, "Numerical and verbal decision analysis: Comparison on practical cases," *Journal of Multi-Criteria Decision Analysis*, vol. 9, no. 6, pp. 263–273, 2000.

[95] O. Larichev, A. Kortnev, and D. Kochin, "Decision support system for classification of a finite set of multicriteria alternatives," *Decision Support Systems*, vol. 33, no. 1, pp. 13–21, 2002.

[96] O. Larichev, A. Mechitov, V. Morgoev, H. Moshkovich, and E. Furems, "Exact duplicates of human judgments," in *Contemporary Issues in Decision Making*, K. Borcherding, O. Larichev, and D. Messick, Eds. Amsterdam: Elsevier, 1990, pp. 43–55.

[97] O. Larichev, H. Moshkovich, and E. Furems, "Decision support system "CLASS"," in *New Directions in Research on Decision Making*, B. Brehmer, H. Jungermann, P. Lourens, and G. Sevón, Eds. Amsterdam: Elsevier, 1986, pp. 303–315.

[98] O. Larichev, H. Moshkovich, E. Furems, A. Mechitov, and V. Morgoev, *Knowledge Acquisition for Construction of the Full and Contradiction Free Knowledge Bases*. Groningen: Iec ProGAMMA, 1991.

[99] O. Larichev and H. Moskovich, "Decision Support System ORCLASS," in *Proceedings of the Tenth International Conference on Multiple Criteria Decision Making*, vol. 4, Taipei, Taiwan, 1992, pp. 341–350.

[100] ——, *Verbal Decision Analysis for Unstructured Problems*. Dordrecht: Kluwer Academic Publishers, 1997.

[101] J. Léger and J. Martel, "A multicriteria assignment procedure for a nominal sorting problematic," *European Journal of Operational Research*, vol. 138, no. 2, pp. 349–364, 2002.

[102] B. Levin, *Combinatorial Engineering of Decomposable Systems*. Dordrecht: Kluwer Academic Publishers, 1998.

[103] J. Marichal, P. Meyer, and M. Roubens, "Sorting multi-attribute alternatives: The TOMASO method," *Computers & Operations Research*, vol. 32, no. 4, pp. 861–877, 2005.

[104] J. Martel and B. Matarazzo, "Other outranking approaches," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston: Springer-Verlag, 2005, pp. 197–264.

[105] R. Massaglia and A. Ostanello, "N-TOMIC: A support system for multicriteria segmentation problems," in *Multiple Criteria Decision Support*, P. Korhonen, A. Lewandowski, and J. Wallenius, Eds. Berlin: Springer-Verlag, 1991, pp. 167–174.

[106] M. Merad, T. Verdel, B. Roy, and S. Kouniali, "Use of multi-criteria decision-aids for risk zoning and management of large area subjected to mining-induced hazards," *Tunnelling and Underground Space Technology*, no. 19, pp. 125–138, 2004.

[107] K. Miettinen, "On the methodology of multiobjective optimization with applications," Ph.D. dissertation, University of Jyväskylä, Jyväskylä, Finland, 1994.

[108] ——, *Nonlinear Multiobjective Optimization*. Dordrecht: Kluwer Academic Publishers, 1999.

[109] B. Mirkin, *Mathematical Classification and Clustering*. Dordrecht: Kluwer Academic Publishers, 1996.

[110] L. Montano-Guzmán, "Fuzzy measures and integrals in the MCDA sorting problematic: Methodology and application to the diagnosis of firms," Ph.D. dissertation, Free Univeristy of Brussels, Brussels, Belgium, 2002.

[111] J. Moscarola and B. Roy, "Procédure automatique d'examen de dossiers fondée sur une segmentation trichotomique en présence de critères multiples," *RAIRO Recherche Opérationnelle*, vol. 11, no. 2, pp. 145–173, 1977.

[112] H. Moshkovich, A. Mechitov, and D. Olson, "Verbal decision analysis," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston: Springer-Verlag, 2005, pp. 609–637.

[113] V. Mousseau, "Eliciting information concerning the relative importance of criteria," in *Advances in Multicriteria Analysis*, P. Pardalos, Y. Siskos, and C. Zopounidis, Eds. Dordrecht: Kluwer Academic Publishers, 1995, pp. 17–43.

[114] V. Mousseau, L. Dias, and J. Figueira, "On the notion of the category size in multiple criteria sorting models," Center for Discrete Mathematics and Theoretical Computer Science, Rutgers, New Jersey, US, Tech. Rep. 2003-02, 2003.

[115] V. Mousseau, L. Dias, J. Figueira, C. Gomes, and J. Clímaco, "Resolving inconsistencies among constraints on the parameters of an MCDA model," *European Journal of Operational Research*, vol. 147, no. 1, pp. 72–93, 2003.

[116] V. Mousseau, J. Figueira, and J. Naux, "Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results," *European Journal of Operational Research*, vol. 130, no. 2, pp. 263–275, 2001.

[117] V. Mousseau, B. Roy, and I. Sommerlatt, "Development of a decision aiding tool for the evolution of public transport ticket pricing in the Paris region," in *A-MCD-A Aide Multicritère à la Décision - Multiple Criteria Decision Aiding*, M. P. A. Colorni and B. Roy, Eds. Luxembourg: Joint Research Center, European Commision, 2001, pp. 213–230.

[118] V. Mousseau and R. Slowinski, "Inferring an ELECTRE TRI model from assignment examples," *Journal of Global Optimization*, vol. 12, no. 2, pp. 157–174, 1998.

[119] V. Mousseau, R. Slowinski, and P. Zielniewicz, "ELECTRE TRI 2.0a : Methodological guide and user's documentation," Université de Paris-Dauphine, Document du LAMSADE 111, 1999.

[120] ——, "A user-oriented implementation of the ELECTRE TRI method integrating preference elicitation support," *Computers & Operations Research*, vol. 27, no. 7-8, pp. 757–777, 2000.

[121] P. Nemery de Bellevaux, "MultiCriteria Clustering," Master's thesis, Free University of Brussels, Brussels, Belgium, 2006.

[122] P. Nemery de Bellevaux and C. Lamboray, "FlowSort: A sorting method based on flows," SMG, Universite' Libre de Bruxelles, Brussels, Belgium, Tech. Rep., 2007.

[123] A. Newell, J. Shaw, and H. Simon, "Chess-playing programs and the problem of complexity," *IBM Journal of Research and Development*, vol. 2, no. 2, pp. 320–335, 1958.

[124] A. Ngo The and V. Mousseau, "Using assignment examples to infer category limits for the ELECTRE TRI method," *Journal of Multi-Criteria Decision Analysis*, vol. 11, no. 1, pp. 29–43, 2002.

[125] M. Norese and S. Viale, "A multi-profile sorting procedure in the public administration," *European Journal of Operational Research*, vol. 138, no. 2, pp. 365–379, 2002.

[126] S. Oh and W. Pedrycz, "Identification of fuzzy systems by means of an auto-tuning algorithm and its application to nonlinear systems," *Fuzzy Sets and Systems*, vol. 115, no. 2, pp. 205–230, 2000.

[127] M. Oral and O. Kettani, "Modelling the process of multiattribute choice," *Journal of the Operational Research Society*, vol. 40, no. 3, pp. 281–291, 1989.

[128] E. Paschetta and A. Tsoukiàs, "A real world MCDA application: Evaluating software," *Journal of Multi-Criteria Decision Analysis*, vol. 9, pp. 205–226, 2000.

[129] Z. Pawlak, *Rough Sets. Theoretical Aspects of Reasoning about Data*. Dordrecht: Kluwer Academic Publishers, 1991.

[130] Z. Pawlak and R. Slowinski, "Rough set approach to multi-attribute decision analysis," *European Journal of Operational Research*, vol. 72, no. 3, pp. 443–459, 1994.

[131] P. Perny, "Multicriteria filtering methods based on concordance/non-discordance principles," *Annals of Operations Research*, vol. 80, pp. 137–167, 1998.

[132] Principia Cybernetica Web. [Online]. Available: http://pespmc1.vub.ac.be/

[133] P. Rietveld, *Multiple Objective Decision Methods and Regional Planning*. Amsterdam: Elsevier, 1980.

[134] R. Robera, J. Penberthy, T. Loboschefski, D. Cox, and B. Kovatchev, "Combined psychophysiological assessment of ADHD: A pilot study of Bayesian probability approach illustrated by appraisal of ADHD in female collage students," *Applied Psychophysiology and Biofeedback*, vol. 29, no. 1, pp. 1–18, 2004.

[135] M. Rogers, M. Bruen, and L. Maystre, *ELECTRE and decision support*. Dordrecht: Kluwer Academic Publishers, 2000.

[136] B. Roy, "A multicriteria analysis for trichotomic segmentation problems," in *Multiple Criteria Analysis: Operational Methods*, P. Nijkamp and J. Spronk, Eds. Farnborough: Gower Press, 1981, pp. 245–257.

[137] ——, "Main sources of inaccurate determination, uncertainty and imprecision," *Mathematical and Computer Modelling*, vol. 12, no. 10/11, pp. 1245–1254, 1989.

[138] ——, "The outranking approach and the foundations of ELECTRE methods," in *Readings in Multiple Criteria Decision Aid*, C. Bana e Costa, Ed. Berlin: Springer-Verlag, 1990, pp. 155–183.

[139] ——, "The outranking approach and the foundations of ELECTRE methods," *Theory and Decision*, vol. 31, pp. 49–73, 1991.

[140] ——, *Multicriteria Methodology for Decision Aiding*. Dordrecht: Kluwer Academic Publishers, 1996.

[141] T. Saaty, *The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation*. New York: McGraw-Hill, 1980.

[142] A. Scarelli and S. Narula, "A multicriteria assignment problem," *Journal of Multi-Criteria Decision Analysis*, vol. 11, pp. 65–74, 2002.

[143] H. Simon, "From substantial to procedural rationality," in *Method and Appraisal in Economics*, S. Latsis, Ed. Cambridge: Cambridge University Press, 1976, pp. 129–148.

[144] ——, *The Sciences of the Artificial*. Cambridge: The MIT Press, 1985.

[145] Y. Siskos, E. Grigoroudis, E. Krassadaki, and N. Matsatsinis, "A multicriteria accreditation system for information technology skills and qualifications," *European Journal of Operational Research*, vol. 182, no. 2, pp. 867 – 885, 2007.

[146] R. Slowinski and D. Vanderpooten, "Similarity relation as a basis for rough approximations," in *Advances in Machine Intelligence and Soft-Computing*, P. Wang, Ed. Durham: Duke University Press, 1997, pp. 17–33.

[147] C. Smith, "Some examples of discrimination," *Annals of Eugenics*, vol. 13, pp. 272–282, 1947.

[148] V. Srinivasan and Y. H. Kim, "Credit granting: A comparative analysis of classification procedures," *Journal of Finance*, vol. 42, no. 3, pp. 665–683, 1987.

[149] H. Taha, *Operational Research: An Introduction*, 6th ed. New Jersey: Prentice-Hall, 1997.

[150] R. Tannock, "Attention Deficit Hyperactivity Disorder: Advances in cognitive, neurobiological, and genetic research," *Journal of the Child Psychology and Psychiatry*, vol. 39, pp. 65–99, 1998.

[151] T. Tervonen, J. Almeida-Dias, J. Figueira, R. Lahdelma, and P. Salminen, "SMAA-TRI: A parameter stability analysis method for ELECTRE TRI," in *Environmental Security in Harbors and Coastal Areas*, I. Linkov, G. Kiker, and R. Wenning, Eds. Berlin: Springer-Verlag, 2007.

[152] T. Tervonen and R. Lahdelma, "Implementing stochastic multicriteria acceptability analysis," *European Journal of Operational Research*, vol. 178, no. 2, pp. 500–513, 2007.

[153] T. Tervonen and J. Figueira, "A survey on Stochastic Multicriteria Acceptability Analysis methods," *Journal of Multi-Criteria Decision Analysis*, 2007, to appear. [Online]. Available: http://www3.interscience.wiley.com/cgi-bin/abstract/117345965/ABSTRACT

[154] T. Tervonen, H. Hakonen, and R. Lahdelma, "Elevator planning with Stochastic Multicriteria Acceptability Analysis," *Omega*, vol. 36, no. 3, pp. 352 – 362, 2008.

[155] T. Tervonen, R. Lahdelma, and P. Salminen, "A method for eliciting and combining group preferences for Stochastic Multicriteria Acceptability Analysis," TUCS - Turku Center for Computer Science," Tech. Rep. 638, 2004.

[156] A. Tsoukiàs, "From decision theory to decision aiding methodology," *European Journal of Operational Research*, vol. 187, no. 1, pp. 138–161, 2008.

[157] A. Valls, "ClusDM: A multiple criteria decision making method for heterogeneous data sets," Ph.D. dissertation, Technical University of Catalonia, Barcelona, Spain, 2002.

[158] P. Vincke, *Multicriteria Decision-Aid*. New York: J. Wiley, 1992.

[159] J. von Neumann and O. Morgenstern, *Theory of games and economic behavior*, 2nd ed. Princeton: Princeton University Press, 1947.

[160] M. Weber and K. Borcherding, "Behavioral influences on weight judgments in multiattribute decision making," *European Journal of Operational Research*, vol. 67, pp. 1–12, 1993.

[161] I. Yevseyeva and K. Miettinen, "Survey of multicriteria decision aiding methods for ordinal classification with some new methodology," University of Jyväskylä, Department of Mathematical Information Technology, Series B. Scientific Computing B 2/2005, 2005.

[162] I. Yevseyeva, K. Miettinen, and P. Räsänen, "Decision support system for Attention Deficit Hyperactivity Disorder diagnostics," in *Proceedings of ORP3: A EURO conference for young OR researchers and practitioners*, C. Maroto, R. Ruiz, J. Alcaraz, E. Vallada, and F. Crespo, Eds., Valencia, Spain, 2005, pp. 183–194.

[163] ——, "Verbal ordinal classification with multicriteria decision aiding," *European Journal of Operational Research*, vol. 185, pp. 964–983, 2008.

[164] I. Yevseyeva, K. Miettinen, and P. Salminen, "SMAA-Classification: A new method for nominal classification," Helsinki School of Economics," Working Paper W-422, 2007.

[165] W. Yu, "Aide multicritere a la de cision dans le cadre de la problematique du tri: Concepts, me thodes et applications," Ph.D. dissertation, University of Paris-Dauphine, Paris, France, 1992.

[166] L. Zadeh, "Fuzzy sets," *Information Control*, vol. 8, pp. 338–353, 1965.

[167] H. Zimmermann, "Fuzzy programming and linear programming with several objective functions," *Fuzzy Sets and Systems*, vol. 1, pp. 45–55, 1978.

[168] C. Zopounidis, "MCDA methodologies for classification and sorting," *European Journal of Operational Research*, vol. 138, no. 2, pp. 227–228, 2002.

[169] C. Zopounidis and M. Doumpos, "Business failure prediction using the UTADIS multicriteria analysis method," *Journal of the Operational Research Society*, vol. 50, no. 11, pp. 1138–1148, 1999.

[170] ——, "PREFDIS: A multicriteria decision support system for sorting decision problems," *Computers & Operations Research*, vol. 27, no. 7-8, pp. 779–797, 2000.

[171] ——, "A preference disaggregation decision support system for financial classification problems," *European Journal of Operational Research*, vol. 130, no. 2, pp. 402–413, 2001.

[172] ——, "Multicriteria classification and sorting methods: A literature review," *European Journal of Operational Research*, vol. 138, no. 2, pp. 229–246, 2002.

APPENDIX 1 DIAGNOSTICS OF ATTENTION-DEFICIT HYPERACTIVITY DISORDER

From Structured Interview for Diagnostic Assessment of Children (SIDAC) for Diagnostic and Statistical Manual for Mental Disorders (DSM-IV) diagnoses of children ages 6-12

Has patient had any of the following problems for at least the last six months?

Inattention:

- Often fails to give close attention to details or makes careless mistakes in schoolwork, chores, or other activities?
- Often has difficulty sustaining attention in tasks or play activities?
- Often does not seem to listen to what is being said to him or her?
- Often does not follow through on instructions and fails to finish schoolwork, chores, or duties at home (not due to oppositional behavior or failure to understand directions)?
- Often has difficulties organizing tasks/activities?
- Often avoids or strongly dislikes tasks (such as schoolwork or homework) that require sustained mental effort?
- Often loses things necessary for tasks or activities (e.g., school assignments, pencils, books, tools, or toys)?
- Is often easily distracted by extraneous stimuli?
- Is often forgetful in daily activities?

Hyperactivity:

- Often fidgets with hands or feet or squirms while seating?
- Leaves classroom or other situations in which remaining seated is expected?
- Often runs about or climbs excessively in situation where it is inappropriate?
- Often has difficulty playing or engaging in leisure activities quietly?
- Is often "on the go" or often acts as if "driven by a motor"?
- Often talks excessively?

Impulsivity:

- Often blurts out answers to questions before the questions have been completed?

- Often has difficulty waiting in lines or awaiting turn in games or group situations?
- Often interrupts or intrudes on others (e.g., butts into conversations or games)?

YHTEENVETO (FINNISH SUMMARY)

Monikriteerisen päätöksenteon lähestymistapoja ja periaatteita voidaan soveltaa luokitteluongelmissa, joissa päätökset riippuvat päätöksentekijän subjektiivisista tiedoista, mieltymyksistä ja preferensseistä. Tällaisten menetelmien päätöksentekoa helpottavat ominaisuudet tarjoavat päätöksentekijälle huomattavaa apua ongelman määrittelyn, muotoilun ja tukimenettelyjen kautta. Tämän työn taustalla on tarve strukturoida monikriteerisen päätöksenteon lähestymistapoja hyödyntävää luokittelumenetelmästöä ja alan problematiikkaa. Työssä esitetään kattava ja ajantasaisen katsaus luokittelumenetelmiin, jotka käyttävät monikriteerisen päätöksenteon lähestymistapoja. Lisäksi olemassaolevat menetelmät jaotellaan ryhmiin ominaisuuksiensa mukaan ja tarkastellaan yleisimmin käytettyjä menetelmiä yksityiskohtaisemmin. Esille tuodaan kunkin menetelmän vahvuuksia ja heikkouksia, sekä" eroja ja yhtäläisyyksiä muihin verrattuna. Tämän analyysin esiintuomat puutteet ovat innoittaneet kahden uuden menetelmän kehittämistä.

Tässä työssä laajennetaan stokastiseen hyvä"ksyttävyysanalyysiin perustuva SMAA-menetelmä nominaaliseen luokitteluun. Tämä menetelmä ei edellytä parametri-informaation olemassaoloa, vaan se olettaa vain, että päätöksentekijä voi osoittaa yhden tai useampia luokitteluesimerkkejä kuhunkin luokkaan. Menetelmä antaa tuloksena tietoa kunkin vaihtoehdon hyväksyttävyydesta" luokiteltavaksi eri luokkiin. Toisaalta taas silloin, kun luokitteluesimerkkejä ei voida osoittaa eri luokille, sanallista päätösanalyysia voidaan hyödyntää ordinaalisessa luokittelussa. Tällöin oletetaan, että luokista tiedetään vain niiden lukumäärä ja järjestys. Nämä menetelmät ovat interaktiivisia. Vuorovaikutteissa prosessissa päätöksentekijää pyydetään luokittelemaan joitakin menetelmän valitsemia vaihtoehtoja, mutta ei kaikkia. Tässä työssä esitellään sanallisen päätösanalyysin periaatteita käyttävä DC-menetelmä ordinaaliseen luokitteluun.

Kehitetyt menetelmät sekä muutamia muita menetelmiä on implementoitu päätöksenteon tukijärjestelmän muotoon. Työssä kehitettyjä menetelmiä havainnollistetaan yksinkertaisin esimerkein. Lisäksi esitetään joitakin numeerisia testituloksia ja vertailuja käyttäen monikriteerisen päätöksenteon kirjallisuudesta löytyviä luokittelutehtäviä. Testien tulokset tukevat kehitettyjen menetelmien tehokkuutta. Uutta DC-menetelmää sovelletaan myös neuropsykologisessa diagnostiikassa, erityisesti ADHD-tapausten diagnosoinnissa. Jotta ajantasainen kuva monikriteerisen päätöksenteon lähetymistapoja hyödyntävistä luokittelumenetelmistä täydentyy, esitetään työssä myös suosituksia kuhunkin luokitteluongelmaan sopivan menetelmän valitsemiseksi.