INFORMATION THEORETIC MEASURE BASED INTERACTIVE APPROACHES TO MULTI-CRITERIA SORTING PROBLEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF SOCIAL SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ALİ ÖZARSLAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
THE DEPARTMENT OF BUSINESS ADMINISTRATION

SEPTEMBER 2021

Approval of the thesis:

INFORMATION THEORETIC MEASURE BASED INTERACTIVE APPROACHES TO MULTI-CRITERIA SORTING PROBLEMS

submitted by **ALİ ÖZARSLAN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Business Administration, the Graduate School of Social Sciences of Middle East Technical University** by,

Prof. Dr. Yaşar KONDAKÇI
Dean
Graduate School of Social Sciences

Prof. Dr. Z. Nuray GÜNER
Head of Department
Department of Business Administration

Assist. Prof. Dr. Gülşah KARAKAYA
Supervisor
Department of Business Administration

Examining Committee Members:

Prof. Dr. Özlem ÖZDEMİR (Head of the Examining Committee)
Middle East Technical University
Department of Business Administration

Assist. Prof. Dr. Gülşah KARAKAYA (Supervisor)
Middle East Technical University
Department of Business Administration

Prof. Dr. Uğur SOYTAŞ
Denmark Technical University
Department of Technology, Management and Economics

Assoc. Prof. Dr. Ceren TUNCER ŞAKAR
Hacettepe University
Department of Industrial Engineering

Assist. Prof. Dr. Özlem KARSU
İhsan Doğramacı Bilkent University
Department of Industrial Engineering

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Ali ÖZARSLAN

Signature:

ABSTRACT

INFORMATION THEORETIC MEASURE BASED INTERACTIVE APPROACHES TO MULTI-CRITERIA SORTING PROBLEMS

ÖZARSLAN, Ali
Ph.D., Department of Business Administration
Supervisor: Assist. Prof. Dr. Gülşah KARAKAYA

September 2021, 132 pages

In this thesis, we develop interactive approaches for sorting alternatives evaluated on multiple criteria. We assume that the preferences of the decision maker are consistent with an additive preference function in general monotone and piecewise linear forms. We progressively solve mathematical models to identify the possible category range of the alternatives and ask the decision maker to place an alternative in each iteration. Based on the mathematical models and Monte Carlo simulations, we hypothetically assign alternatives to find the assignment frequency and the probability of an alternative to be assigned to a category. We then use an information theoretic measure, relative entropy, in the determination of the assignment uncertainties and the selection of the alternative that will be assigned to a category by the decision maker. In our non-probabilistic approach, our algorithm guarantees the assignment of all available alternatives to their true categories assuming that the preferences of the decision maker are consistent with an additive function. Our probabilistic algorithm allows the assignment of the alternatives based on the estimated assignment probabilities once the decision maker provides enough assignment information. We implement the proposed algorithms and different benchmark algorithms on three example problems from the literature as well as randomly generated problems. We consider the cases with/without category size restrictions and initial assignments in problem settings. The results show that the proposed algorithms perform well in terms of decreasing the cognitive burden of the decision maker, decreasing the misclassification of the alterantives and the length of the decision process.

Keywords: Multiple Criteria Sorting; Additive Preference Function; Mathematical Programming; Relative Entropy; Category Size Restriction

ÖZ

ÇOK KRİTERLİ SINIFLANDIRMA PROBLEMLERİNE BİLGİ TEORİK ÖLÇÜ TABANLI ETKİLEŞİMLİ YAKLAŞIMLAR

ÖZARSLAN, Ali
Doktora, İşletme Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi Gülşah KARAKAYA

Eylül 2021, 132 sayfa

Bu tezde çoklu kriterlere göre değerlendirilen alternatifleri sınıflandırmak için etkileşimli yaklaşımlar geliştirilmektedir. Karar vericinin tercihlerinin genel monoton ve parçalı doğrusal formlarda toplamsal bir tercih fonksiyonu ile tutarlı olduğu varsayılmaktadır. Alternatiflerin olası kategori aralığını belirlemek için iterasyonlar boyunca karar vericiden atama bilgisi alarak matematiksel modeller çözülmektedir. Alternatiflerin atanma sıklığını ve olasılıklarını bulmak için matematiksel modeller ve Monte Carlo simülasyonları yoluyla farazi atama yapılmaktadır ve alternatiflerin kategorilere atanma olasılığı bulunmaktadır. Atama belirsizliklerinin hesaplanmasında ve karar verici tarafından atanacak alternatiflerin seçiminde bir bilgi teorik ölçüsü olan göreceli entropi kullanılmaktadır. Geliştirilen olasılıksız algoritmada karar vericinin tercihlerinin toplamsal fayda fonksiyonu ile tutarlı olduğu durumda tüm alternatiflerin doğru kategorilerine atanması garanti edilmektedir. Geliştirilen olasılıksal algoritma karar vericiden yeterli atama bilgisi sağladığında atama olasılıklarına göre alternatiflerin atanmasına izin vermektedir. Önerilen ve kıyaslama yapılacak algoritmaların performanslarını ölçmek için literatürden üç örnek problem ve rassal olarak oluşturulmuş problemler üzerinde uygulama yapılmaktadır. Problemlerde başlangıç atamaları ve kategori büyüklüğü kısıtlamaları olan/olmayan durumlar dikkate alınmaktadır. Sonuçlar önerilen algoritmaların karar vericinin bilişsel yükünü ve sınıflandırma hatalarını azaltmada ve karar verme sürecini kısaltmada iyi performansa sahip olduğunu göstermektedir.

Anahtar Kelimeler: Çok Kriterli Sınıflandırma; Toplamsal Tercih Fonksiyonu; Matematiksel Programlama; Göreceli Entropi; Kategori Büyüklüğü Kısıtlaması

To My Family

ACKNOWLEDGMENTS

I would like to gratefully acknowledge various people who supported this study. First and foremost, I would like to thank my advisor, Asst. Prof. Dr. Gülşah Karakaya, for her guidance, patience and endless support. I appreciate all her contributions of time, ideas and effort to make my Ph.D. experience productive. Her approachability and responsiveness have always been a big support in completing this thesis. She has not only guided me throughout my dissertation, but also encouraged me to work on different topics.

I would like to thank my defense committee members Prof. Dr. Özlem Özdemir, Prof. Dr. Uğur Soytaş, Assoc. Prof. Dr. Ceren Tuncer Şakar and Asst. Prof. Dr. Özlem Karsu for their time, interest, insightful questions and helpful comments.

I would like to express my special appreciation and thanks to Assoc. Prof. Dr. Mustafa Sinan Gönül and Prof. Dr. Ramazan Sarı for their contribution to the Ph.D. program.

These acknowledgments would not be complete without mentioning my friends, Mustafa Fırat, Hatice Gonca Bulur, and Ecem Gizem Hüner who have always supported and encouraged me during my Ph.D. study.

My deepest appreciation belongs to my wife, my daughter, and my son for their endless support and unconditional love.

TABLE OF CONTENTS

APPENDICES

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Decision problems in any real or hypothetical setting usually require the decision maker (DM) to consider decision units or alternatives. Such problems are referred to as Multiple Criteria Decision Making (MCDM) problems when the alternatives are evaluated on multiple criteria. In most cases, the criteria used to evaluate the alternatives conflict with each other. For instance, consider the criteria, quality and price, to evaluate the suppliers in a supplier selection problem. Since the price of a product with high quality is generally expected to be high and the price of a product with less quality to be low, these two criteria can be considered as conflicting criteria. Another example is the usage of return and risk by the investors when deciding on the financial instruments to invest in.

In case of conflicting criteria, the DM needs to make judgment about the trade-offs between such criteria. These judgments provide information about the preference structure of the DM. Multiple Criteria Decision Aid (MCDA) methods utilize the preferences of the DM to assist in the decision making process. The aim in these methods is to derive a way to reach a solution by facilitating an understanding of the preferences of the DM through eliciting decision examples.

In most decision problems, there are discrete number of available alternatives to be evaluated on multiple criteria. Roy (1981) divides the MCDM problems into four as choice, ranking, classification/sorting and description problems as shown in Figure 1. The first three types of problems require evaluation of the alternatives to end up with a specific outcome whereas the last type of problems, description problems, are encountered in situations where the alternatives and their unique features are described to define the characteristics of the problem. The aim in choice problems is to select the most preferred alternative or a subset of incomparable or equivalent alternatives. A typical example for choice problems is to identify the most suitable candidate(s) for a position in a company. Another example is to select the most appropriate supplier(s) for purchasing purposes.

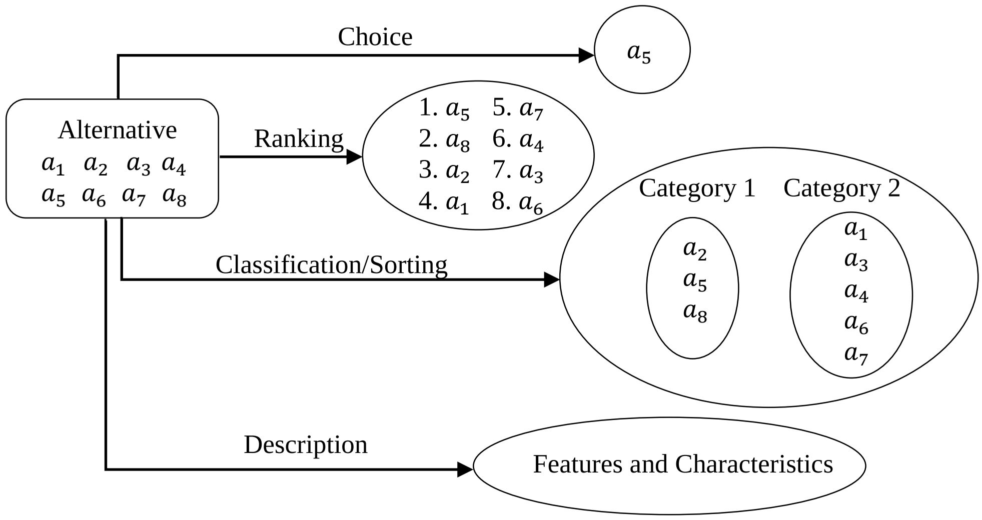


Figure 1. MCDM problem types (Source: Roy, 1981)

The second type is the ranking problems where the alternatives are listed in a rank from the most preferred to least preferred ones. Each alternative included in the ranking has a preferential relationship with the rest of the alternatives in a typical ranking output. The most preferred alternatives are usually the ones with the highest rankings. The ranking of the two alternatives can be identical when the DM is indifferent between these alternatives. The rankings are called partial (full) when the ties between the rankings are (not) allowed. Country rankings based on energy sustainability and university rankings based on academic excellence are the typical examples of ranking problems.

In classification/sorting problems, the DM is required to assign alternatives into two or more categories. The difference between the classification and sorting problems is that categories are nominal in classification problems whereas they are ordinal in sorting problems. In a typical classification problem, alternatives are assigned to undefined or predefined categories based on similar characteristics shaped by the category members. The statistical methods such as cluster analysis (Driver and Kroeber, 1932; Zubin, 1938; Tryon, 1939) and multi-dimensional scaling (Torgerson, 1952; 1958; Gower, 1966) are usually used for classification problems. The classification of the species or chronic diseases based on multiple criteria are the examples in this field.

In sorting problems, the aim is to assign the alternatives to the categories in such a way that an alternative in a better category is preferred to all alternatives in worse categories. In most cases, the categories are separated from each other with boundaries such as threshold values. There are several situations or areas in daily life and business context where the sorting problems are encountered. A representative example is to assign the master applicants or loan applications into three categories as accepted, pending, and rejected. Assignment of firms into categories based on their risk attitudes is another example for sorting problems typically performed by the credit rating agencies.

There are different approaches developed to solve Multi-Criteria Sorting (MCS) problems. These approaches can be categorized as multivariate statistical analysis, non-parametric statistical learning approaches and Preference Disaggregation Analysis (PDA). Well-known multivariate statistical methods are logit (Bliss, 1934), probit (Berkson, 1944) and discriminant analysis (Fisher, 1936; Smith, 1947). These techniques are usually parametric approaches restricted with statistical assumptions such as normality and homoscedasticity. Non-parametric methods are neural networks (Östermark, 1999), support vector machines (Kartal et al. 2016), fuzzy set theory (Belacel and Boulassel, 2004) and rough sets (Greco et al., 2001; 2002).

The aim of PDA is to build a preference model that is consistent with preferences of the DM (Jacquet-Lagreze and Siskos, 2001). PDA utilizes the decision examples of the DM to explore the parameters of the preference model. One way to use the DM's preferences is to ask him/her to specify the values of model parameters required to assign the alternatives. This is called direct elicitation technique and is usually cumbersome in terms of cognitive burden on the DM. In PDA, typically, outranking-based approaches are considered within the direct elicitation techniques. ELECTRE-TRI is a widely used outranking-based method for MCS problems (Yu, 1992). The method requires the DM to specify weights, thresholds, and representative category profiles. By this way, the outranking relations in form of pairwise comparisons between the alternatives are defined.

As an alternative to the direct elicitation techniques, function-based approaches are considered in PDA where the preference structure of the DM is usually assumed to have an implicit function which is called a preference function in general. Function-based approaches are indirect elicitation techniques suggested to approximate the model parameters based on the decision examples obtained from the DM. The most common decision example in approaches to MCS problems is the assignment information of a set of reference alternatives. The DM may initially provide the assignment examples (e.g., "alternative a belongs to the first category") and the rest of the alternatives are assigned by the decision model. Another way of obtaining assignment examples is to progressively elicit category information through interaction with the DM. In interactive approaches, the involvement of the DM in decision process facilitates better understanding and incremental learning of the model parameters by him/her.

Several sorting approaches have been developed assuming that the preferences of the DM are consistent with different forms of additive preference functions. Additive preference function in monotonically non-decreasing form is the general preference structure where the local utilities in each criterion are added to find the aggregate utility of the alternatives. Piecewise linear (Köksalan and Özpeynirci, 2009) and quasiconcave preference functions (Ulu and Köksalan, 2001; 2014) are widely used forms of additive functions to represent the preferences of a DM. Besides, Tchebycheff (Soylu, 2011), quadratic (Özpeynirci et al., 2017) and different forms of L_p norm (Çelik et al., 2015) functions are utilized to demonstrate the preference structure of a DM in MCS problems.

In this thesis, we develop interactive sorting approaches assuming that the preferences of the DM are consistent with an additive preference function. Considering piecewise linear and general monotone preference functions, we solve mathematical models to define the category range of the alternatives. At each iteration, an alternative is assigned by the DM and we incorporate this information to the mathematical models in order to narrow down the possible assignment of the alternatives. We hypothetically assign the alternatives to the categories in several times based on a set of compatible parameters obtained from mathematical models and Monte Carlo simulations. By this way, we gather information about the assignment frequencies of the alternatives to the categories. The previous studies do not consider the parameters of the mathematical models for hypothetical assignments. Moreover, the previous simulation-based approaches do not take into account the incompatibility of the randomly generated parameters while we propose a practical approach to minimize the incompatibility problem.

We convert the frequency information to probability of belonging to a category for each alternative. We use an information theoretic measure, relative entropy, to measure the assignment uncertainty of the alternatives. Although there are some approaches that utilize entropy calculation in multi-criteria ranking problems, to the best of our knowledge, there has been no previous work that engages relative entropy in measuring uncertainty and ambiguity within the sorting framework. This study develops an efficient method to shorten the decision process in the assignment of alternatives and decrease the cognitive burden of the DM.

We utilize the assignment uncertainties in selecting the alternative to ask the DM and identifying the uncertainty of the system. We consider non-probabilistic and probabilistic assignments as well as the cases with/out the category size restrictions. In our non-probabilistic algorithm, alternatives are assigned to their true categories by the DM and the mathematical models. Our probabilistic algorithm allows the probabilistic assignment of the alternatives once the DM provides sufficient assignment information. As far as we know, there is no interactive study that makes probabilistic assignments except one study. In that study, it is observed that the misclassifications are at a high level. This gap in the literature is tried to be filled with the proposed interactive probabilistic method which aims to complete the decision process with the least information to be obtained from DM and the least classification error.

We test the performances of our non-probabilistic and probabilistic algorithms on three example problems from the literature as well as randomly generated problems by comparing with benchmark algorithms and report our findings. The rest of the thesis is organized as follows: In Chapter 2, we review the literature on MCDA approaches and provide an overview of the methods and concepts utilized in the proposed algorithms. In Chapter 3, we first present the proposed interactive approaches and then explain the benchmark approaches that are used for comparison. In Chapter 4, we conduct computational experiments to test the approaches on several problems. We finally present the concluding remarks and future research directions in Chapter 5.

CHAPTER 2

LITERATURE REVIEW AND BACKGROUND

In this chapter, we first give a brief summary of the approaches to multiple criteria choice and ranking problems. Then, we explain the approaches to MCS problems in detail. Lastly, we provide an overview of the methods and concepts utilized in the analysis.

2.1 Approaches to multiple criteria choice and ranking problems

Multiple criteria choice problems and ranking problems share common characteristics. It is typical for both types of problems to require judgments of the DM in form of pairwise comparisons of the alternatives. Furthermore, the solution may change depending on the set of alternatives included in the problem. For example, the most preferred alternative or the ranking of an alternative may change when new alternatives become available. Another similarity is that the outcome of a ranking approach provides the most preferred alternative(s) as well. For these reasons, there are MCDA methods that are used to handle both choice and ranking problems, and these methods can be listed as: AHP (Saaty, 1980), SMART (Edwards, 1971), SMARTER (Edwards and Barron, 1994), ORESTE (Pastijin and Leysen, 1989), UTA (Jaquet Lagreze and Siskos, 1982), MACBETH (Bana e Costa et al., 2005), PROMETHEE (Brans et al., 1986), ELECTRE (Roy, 1968; 1991), TOPSIS (Hwang and Yong, 1981) and VIKOR (Opricovic, 1998).

The aforementioned MCDA methods for choice and ranking problems are developed based on two main approaches: value or utility-based criteria aggregation and outranking relations. The former assigns an overall aggregating score for alternatives based on the marginal utilities in each criterion. AHP, UTA, MACBETH, TOPSIS and VIKOR are among the well-known methods considered in this category. These methods are based on Multi-Attribute Utility Theory (MAUT) developed by Keeney and Raiffa (1976). In MAUT, the aim is to map the alternatives evaluated by multiple criteria into a single scale so that they can be compared in terms of overall utilities. The basic principle behind this approach is shown in (2.1) and (2.2). Without loss of generality, suppose that more is better in each criterion. The overall utility of alternative a, $U(a)$, is greater than that of alternative b if and only if alternative a is preferred over alternative b as in (2.1). The overall utilities of alternative a and b are equal when there is a preferential indifference among the alternatives as in (2.2).

$$U(a) > U(b) \leftrightarrow a \succ b \tag{2.1}$$

$$U(a) = U(b) \leftrightarrow a \sim b \tag{2.2}$$

The second type of methods are based on outranking relations between alternatives. The outranking-based methods require the DM to identify weights and thresholds to explore the outranking relations. By this way, the pairwise comparisons between the alternatives are made through binary relation S as in (2.3) (Roy, 1991). Binary relation between alternatives a and b indicates that alternative a outranks alternative b if and only if alternative a is at least as good as alternative b.

$$a\,S\,b \;\leftrightarrow\; a \text{ is at least as good as } b \tag{2.3}$$

PROMETHEE and ELECTRE methods are the well-known outranking-based techniques for choice and ranking problems. PROMETHEE I is designed for partial rankings whereas PROMETHEE II is applied to problems requiring full rankings (Behzadian et al., 2010). ELECTRE I is used in choice problems and ELECTRE II, III and IV are usually applied in ranking problems (Ishizaka and Nemery, 2013).

Data Envelopment Analysis (DEA), introduced by Charnes et al. (1978), evaluates decision making units (DMUs) in terms of efficiency in converting inputs/outputs to outputs/inputs. The efficiency scores of DMUs are calculated by the ratio of weighted sum of outputs to that of inputs where the weights are obtained by mathematical programming technique. DMUs with the relative efficiency score of one are called efficient DMUs while the others are inefficient. Sinuany-Stern et al. (1994) suggested using the D_k measure which is the minimum number of efficient DMUs to be eliminated so that an inefficient DMU can become efficient. Doyle and Green (1994) developed the cross-efficiency scores method for ranking problems. Each DMU is assessed with other DMUs' optimal weights and then the average of the D_k scores is denoted as the cross-efficiency score. Köksalan and Tuncer (2009) addressed the drawbacks of the cross-efficiency scores in case of outliers and crowding of the alternatives in specific areas. The authors proposed the area of efficiency score graph method that considers the converging speed of efficiency score to one when other DMUs are deleted in an order through an integer programming model.

Interactive approaches have been developed for both choice and ranking problems. In these approaches, the preferences of the DM are assumed to be consistent with an underlying preference function. Typically, the search space is reduced in an iterative way utilizing the properties of the assumed preference function. The DM provides preference information between alternatives in a progressive way. The underlying preference function is represented by linear (Zionts, 1981), quasiconcave (Korhonen et al., 1984), quasiconvex (Köksalan et al., 1984), general monotone (Köksalan and Sagala, 1995a) and different forms of L_α (Karakaya et al., 2018) functions in approaches for choice problems.

The convex cone approach of Korhonen et al. (1984) is incorporated in other studies that are based on quasiconcave preference functions. The idea behind the convex cone approach is that the preference cones, constructed by the preferences of the DM and the properties of the quasiconcave functions, can be used to explore additional preferences regarding the other alternatives. Using the preferences of the DM in form of pairwise comparisons, the preference cones are generated by linear programming (LP) models to evaluate the alternatives in terms of cone dominance (see Karsu, 2013 for a review on convex cone approaches).

The aforementioned interactive approaches to choice problems consider the pairwise comparison of subsets of alternatives to find the most preferred one and hence they are not designed for ranking problems. Korhonen and Soismaa (1981) develop an interactive ranking approach based on a linear preference function. In order to rank the alternatives, the weights of criteria are estimated by LP models. Karsu et al. (2018) consider the quasiconcave preference function combined with equity or fairness concerns leading to non-additive preferences on criteria. The authors develop an interactive ranking approach based on the convex cones and polyhedra cones as well as the generalized Lorenz dominance which is widely used in case of equity concerns. Tezcaner Öztürk and Köksalan (2019) develop a framework for interactive elicitation of pairwise comparison information for linear and quasiconcave preference functions. The results show that progressive elicitation of the preferences yields better ranking performance than obtaining a priori preference information.

2.2 Approaches to MCS problems

Sorting problems have different characteristics from choice and ranking problems (Vetschera et al., 2010). In choice and ranking problems, the addition of the new alternatives may change the current position of the available alternatives. In sorting problems on the other hand, the addition of new alternatives does not change the category of the previously assigned alternatives (Zopounidis and Doumpos, 2002). That is why the sorting problems usually require the DM to make absolute judgments while the judgments are relative in choice and ranking problems. Furthermore, in sorting problems to construct the decision model usually assignment examples are utilized rather than the pairwise comparisons between the alternatives as utilized in choice and ranking problems.

Doumpos and Zopounidis (2011) distinguish the widely used MCDA approaches into two parts: statistical learning and preference disaggregation analysis (PDA). Rule-based models are popular among the statistical learning techniques for sorting problems. Greco et al. (2002) develop a rule-based model based on the dominance-based rough set approach (DRSA) of Greco et al. (2001) which is an extension to the rough set theory of Pawlak (1982). In rule-based models, "if *conditions* then *response*" type decision rules are developed to explore the preferences of a DM. Rough approximations based on dominance relations are built to define the upward and downward unions of the categories. The decision rules in form of certain, possible and approximate knowledge provide information about the preferences of the DM which in turn is used for the assignment of alternatives. Greco et al. (2004) show that rule-based models are more useful than outranking-based and function-based approaches in terms of handling the inconsistencies among the decisions of the DM.

2.2.1 Approaches based on Preference Disaggregation Analysis (PDA)

PDA involves construction of the preference model on criteria aggregation based on the preferences of the DM (Jacquet-Lagreze and Siskos, 2001). The general framework of PDA for MCDM problems is illustrated in Figure 2. There have been several applications of PDA in real-world MCDM problems such as financial management (Zopounidis et al., 2000), marketing (Mihelis et al., 2001) and job evaluation (Spyridakos et al., 2001). Doumpos and Zopounidis (2011) mention the two popular paradigms on PDA which are outranking-based and function-based approaches. Recently there has been a growing interest among approaches to MCS problems based on outranking and function-based models.

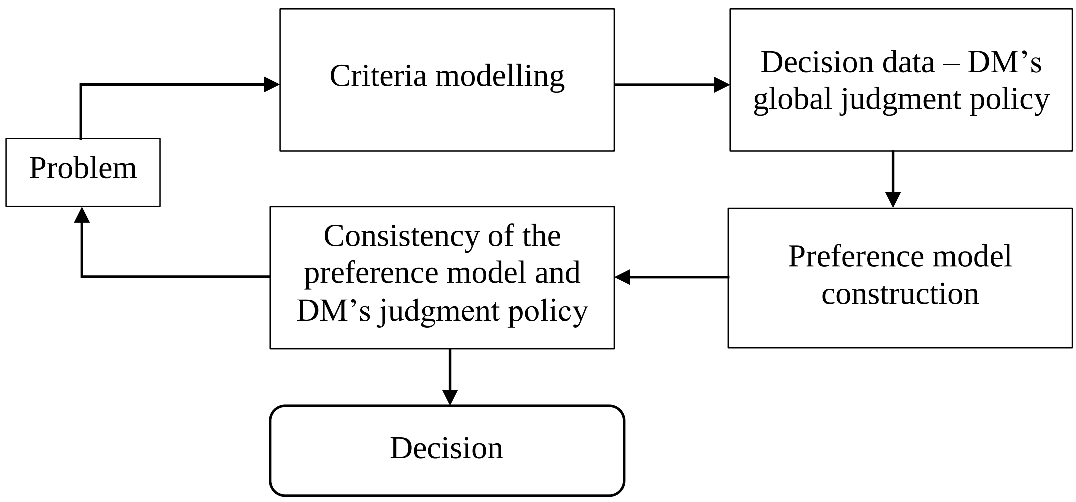


Figure 2. PDA in MCDM problems (Source: Siskos and Spyridakos, 1999)

2.2.1.1 Outranking-based approaches

The general framework for outranking-based approaches is given in Figure 3. Outranking-based sorting approaches usually require the DM to specify weights, thresholds and fictitious profiles for the lower and upper bounds of the categories. By this way, outranking relations are identified and the pairwise comparisons between the alternatives and category profiles are made for sorting alternatives. The ELECTRE TRI method, proposed by Yu (1992) is a well-known sorting method based on outranking relations. In ELECTRE TRI, outranking relations are constructed by concordance and discordance indices that are calculated by the weights of criteria. The possible assignments of the alternatives are identified by the pessimistic and optimistic rules.

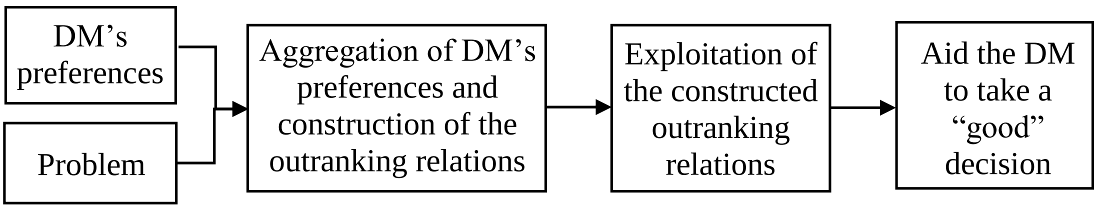


Figure 3. Outranking-based approach (Source: Siskos and Spyridakos, 1999)

The direct elicitation of parameter values from the DM in ELECTRE TRI is widely criticized since it is unrealistic for the DM to simultaneously provide necessary parameters (Dias et al., 2002). Mousseau and Slowinski (1998) develop an interactive approach to obtain assignment examples in a progressive way. The parameters of ELECTRE TRI are estimated through mathematical models based on past assignments of the DM. Majority rule sorting (MR-Sort), developed by Bouyssou and Marchant (2007a; 2007b), is a simplified version of ELECTRE TRI where the category profiles and criteria weights are omitted by eliciting assignment examples from the DM. The recent work of Ramezanian (2019) in ELECTRE-TRI obtains a priori assignment information from the DM and solves mathematical models to define the category profiles with maximum reliability of outranking relations. Tervonen et al. (2009) develop SMAA-TRI where the profiles, thresholds and weights in the ELECTRE-TRI are estimated by Monte Carlo simulations. The study suggests to use uncertain values for the category profiles and thresholds while the weights can be given as intervals.

2.2.1.2 Function-based approaches

One of the widely used indirect elicitation techniques for MCDM problems is to represent the preferences of the DM with an implicit function which is called a preference function. A general framework of the function-based approaches is given in Figure 4 by Siskos and Spyridakos (1999). The value or utility of the DM is assumed to be consistent with a preference function. Additive preference function is widely used in practice for preference modeling (Keeney and Raiffa, 1976). Additive functions have usually monotonically non-decreasing form in cases such as price of a product or expected duration of a service (Ghaderi et al., 2017). Such function is called general monotone which covers the special forms of additive preference functions such as linear, quasiconcave or Tchebycheff functions. Köksalan and Sagala (1995b) develop an interactive approach to test the form of the preference function that is consistent with the preferences of the DM. They progressively search for consistency of the DM's preferences with respect to linear, quasiconcave, quasiconvex or a general monotone additive preference function. In each iteration, the DM is asked to make pairwise comparisons between the alternatives and this information is used to check the consistency of the functions.

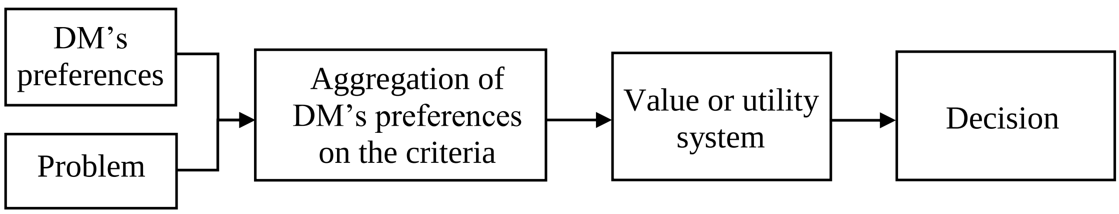


Figure 4. Function-based approaches (Source: Siskos and Spyridakos, 1999)

Some studies argued that a preference function assumption is unrealistic. The preferential independence assumption of MAUT requires that a criterion does not affect the preference order of the alternatives evaluated on another criterion. This assumption is found to be unrealistic since there is often an interaction between criteria (Grabisch 1995; Hillier, 2000). This raises questions regarding the additivity of the functions. One way to allow the interaction between criteria is to use Choquet integral as an evaluation function to represent the preferences of the DM (Benabbou et al., 2017). In order to incorporate non-additivity in a model, Choquet integral values are calculated by defining a capacity function that defines capacity values for each combination of criteria. However, the specifications of Choquet capacities require to define complex fuzzy measures (Grabisch 1995; Tzeng and Huang, 2011; p. 4). On the other hand, additive function-based approaches have been widely utilized in practice due to ease of use and transparency (Doumpos and Zopounidis, 2004).

Linear additive preference function is the simple and practically useful function to represent the preferences of the DM. The weights of criteria are multiplied with the corresponding criteria scores and then summed to find the aggregate utility of an alternative. One of the most common and basic approaches on linear preference function is using fixed weights in each criterion. The weights are usually determined by the authorities or experts; and then aggregate utility of alternatives is calculated by weighted sum of the criteria scores. For instance, Global MBA Program rankings are annually published by Financial Times (FT) and there are 20 criteria such as alumni salary and portion of the women faculty members. FT assigns fixed criteria weights which are criticized by the authorities (Devinney et al., 2008). Köksalan et al. (2010) suggest using weight ranges instead of fixed weights. They show that a little deviation of fixed weights causes to dramatic changes of the rankings which raises questions about the robustness of the methodology.

Quasiconcave preference functions fit well to human behavior in most real-world situations (Arrow and Enthoven, 1961). Principles in economics such as risk aversion in investments and diminishing marginal rate of substitution in consumption are explained by the assumption of quasiconcavity (Silberberg and Suen, 2001, p. 260-261). Quasiconcavity of a function can be approximated by a linear function in a piecewise form in order to avoid to employ a nonlinear function (Zangwill, 1967). The piecewise linear utility function can be used in many situations since it is possible to approximate any nonlinear utility function by this form.

A well-known sorting method, UTADIS was first presented by Devaud et al. (1980) and developed by Jacquet-Lagreze (1995) as a variant of the UTA method of Jacquet-Lagreze and Siskos (1982). In UTADIS the alternatives are assigned to categories by estimating an additive preference function in a piecewise linear form. An LP model is solved to minimize the classification errors due to misclassification of the alternatives. UTADIS is a threshold-based sorting method where the categories are separated by some threshold values. Assignment information of a set of reference alternatives are employed to the LP model to estimate the decision parameters which are in turn used to assign the non-reference set of alternatives, i.e., the remaining alternatives that are not included in the set of reference alternatives. A detailed explanation of UTADIS is given in Section 2.3.1.

UTADIS has been widely applied for real-world problems such as country classification based on energy intensity (Diakoulaki et al., 1999), financial distress prediction of the firms (Zopounidis and Doumpos, 1999), and supplier classification (Manshadi et al., 2015). Zopounidis and Doumpos (1999) compare UTADIS with discriminant analysis, logit and probit analysis; and showed that UTADIS always outperforms the statistical methods. Ulucan and Atıcı (2013) apply UTADIS to perform the country risk classification by using financial data of rating agencies. The authors develop an extension of UTADIS where the classification errors are defined by assignments of the alternatives into multiple categories instead of a single category as in UTADIS. Hence, a goal programming model is solved to minimize total classification errors. The classification results in different datasets including the financial data show that the proposed method outperforms UTADIS in terms of misclassification rates.

UTADIS estimates a single set of parameters that is aimed to fit well to the preference information obtained from the DM. However, there can be several compatible sets of parameters that result in different classification of the non-reference alternatives (Köksalan and Özpeynirci, 2009). The robustness of the decision parameters derived from a single preference function have been taken into consideration within ordinal regression framework (Figueira et al., 2009; Greco et al., 2008; 2010). The role of ordinal regression in function-based approaches is to formulate robust conclusions based on the preferences of the DM. Ordinal regression technique considers the whole set of compatible preference functions instead of a single preference function compatible with the decision examples of the DM as in UTA and UTADIS methods.

2.2.1.2.1 Interactive function-based approaches

Interactive approaches usually assume that the preference function of the DM is in implicit form (Korhonen et al., 1992). Hence, these approaches consider the whole set of compatible preference functions (Greco et al., 2010). A progressive elicitation of the preference information through interaction with the DM helps to enhance learning about the preference structure of him/her. The progressive nature of the approach enables the DM to turn back to the previous steps to reconsider the judgments especially in case of inconsistencies.

One of the pioneer approaches to MCS problems considering the whole set of compatible preference functions is the work of Ulu and Köksalan (2001). The study develops interactive approaches that assign alternatives into two categories - acceptable and unacceptable - assuming that the preferences of the DM are consistent with linear, quasiconcave or general monotone preference function. At each iteration, the DM is asked to assign an alternative and this information is imposed to the mathematical models. In order to determine the category of the alternatives in linear preference function case, LP models are solved by incorporating the assignment examples of the DM as linear constraints. For instance, if the DM assigns alternative a_i to the acceptable category and alternative a_k to the unacceptable category, then the constraint in (2.4) is added to the LP models where w refers to the criteria weight vector. (2.4) enforces the model to assign weight values in such a way that the aggregate utility of a_i should be at least as much as that of a_k.

When defining the category of alternative a_r, constraints (2.5) and (2.6) are added one by one and the feasibility of the model is checked. If the model with constraint in (2.5) is infeasible, then there is no feasible weight vector that makes aggregate utility of a_r better than a_k; thus, a_r belongs to the unacceptable category. The same inference is made for searching the feasibility of the model with constraint (2.6) to check whether or not alternative a_r can be assigned to the acceptable category. The study utilizes the dominance relationships in defining the categories of the alternatives to decrease the number of LP models solved.

$$w(a_i - a_k) \geq 0 \tag{2.4}$$
$$w(a_r - a_k) \geq 0 \tag{2.5}$$
$$w(a_i - a_r) \geq 0 \tag{2.6}$$

In quasiconcave preference function case, Ulu and Köksalan (2001) utilize convex combinations of the alternatives, convex cone approach of Korhonen et al. (1984), and the dominance relations. If an alternative dominates any convex combination of alternatives in the acceptable category, then this dominating alternative is also assigned to the acceptable category. Convex cones are generated between the alternatives in the acceptable category and the non-dominated alternatives in the unacceptable category. If an alternative is inefficient with respect to such convex cones, then this alternative is assigned to the unacceptable category. The authors simply use the dominance relations to determine the assignments in their algorithm for general monotone preference functions since this form utilizes weaker properties when compared to the linear and quasiconcave functions. Ulu and Köksalan (2014) extend the approach of Ulu and Köksalan (2001) when there are more than two categories for underlying quasiconcave preference function.

Köksalan and Ulu (2003) generalize the approach of Ulu and Köksalan (2001) to assign the alternatives into more than two categories assuming an underlying linear preference function for the DM. The possible categories of the alternatives are defined in terms of the best and worst categories. The best and worst possible categories of an alternative is defined as C_1 and C_q, respectively for a q-category problem at the beginning of the algorithm. The aggregate utility of an alternative is compared with that of convex combination of the alternatives assigned to the same category as in (2.7) and (2.8) where $\mu_i \geq 0$ and $\sum_{a_i \in C_k} \mu_i = 1$. If maximization of ε with the constraint in (2.7) results in positive objective function value (ε), then a_j is assigned to C_k or a less preferred category and hence the best possible category of a_j is defined as C_k. Similarly, in the same maximization problem, a positive objective function value in case of constraint in (2.8) indicates that the worst possible category of a_j is C_k. By this way, the category range of the alternatives are narrowed down to find the true categories alternatives belong to.

$$w\left(\sum_{a_i \in C_k} \mu_i a_i - a_j\right) - \varepsilon \geq 0 \tag{2.7}$$

$$w\left(\sum_{a_i \in C_k} \mu_i a_i - a_j\right) + \varepsilon \leq 0 \tag{2.8}$$

Köksalan and Özpeynirci (2009) develop a threshold-based interactive sorting approach using mixed integer programming models (MIPs). They assume that the DM has additive utility function in a piecewise form. In MIPs, each unlabeled alternative is assigned to a category through binary variables. At each iteration, alternative a_t is randomly selected and its overall utility, $U(a_t)$, is compared to category threshold, u_k, as in (2.9) and (2.10). Here, u_k defines the lower bound of category C_k where C_1 corresponds to the most preferred category. The objective in MIPs is to maximize a nonnegative ε. The first model includes constraint (2.9) whereas the second model contains constraint (2.10). Obtaining an infeasible solution in the first model indicates that the worst possible category of alternative a_t is C_k. If the second model is infeasible, then C_k is the best category of alternative a_t. By this way, the possible category range of alternatives are narrowed down at each iteration. Then, the selected alternative is presented to the DM with its category range unless its exact category is found. The DM places the alternative into a category and this information is incorporated to the model in the following iterations. The assignments are done either by the model or the DM until all alternatives are assigned.

$$U(a_t) \leq u_k - \varepsilon \tag{2.9}$$

$$U(a_t) \geq u_{k-1} \tag{2.10}$$

The usage of mathematical programming models to identify the possible assignments of the alternatives is generalized in the robust ordinal regression (ROR) principle (Greco et al., 2008). ROR principle addresses all possible sets of parameters compatible with the preferences of the DM. Greco et al. (2010) adapt the ROR principle by considering the set of general monotone additive utility functions rather than piecewise linear form. The UTADIS^{GMS} method is developed to classify alternatives considering the set of compatible additive utility functions. Based on the assignment examples provided by the DM, the alternatives are necessarily or possibly assigned to one or several contiguous categories with respect to the set of compatible value functions derived from the preference relations of the assignments of the alternatives. The method is suggested to be used as an interactive approach to sorting problems. At each iteration, the possible and necessary categories of the alternatives are updated and presented to the DM.

Benabbou et al. (2017) develop an interactive regret-based approach based on Choquet integral for threshold-based sorting problems. The Choquet integrals allow interacting criteria and non-additive preferences (Choquet, 1955; Denneberg, 1994). The study assumes that the thresholds that separate the categories are defined by the DM at the beginning of the decision process. The regret approach considers the incorrect assignment of the alternatives based on the difference between thresholds and the Choquet values. LP models are solved in order to find the assignment that minimizes the regret due to misclassification. The minimax regret strategy is suggested to be used in the information gathering process from the DM.

Another interactive approach with predefined thresholds is the recent study of Kang et al. (2020) which is based on linear preference function. In order to establish linear relations between pairs of criteria weights, the authors generate hypothetical alternatives to ask the DM to make comparisons in a progressive way. The rankings of the decision weights are obtained by the preferences of the DM among the hypothetical alternatives. The rankings are incorporated to the LP models to find the possible categories of the alternatives at each iteration.

2.2.1.2.2 Approaches for constrained sorting problems

In some sorting problems, the DM may initially provide information about his/her preferences or some limitations due to nature of the problem. For instance, the human resources manager may indicate that experience-based criteria are more important than education-based criteria when classifying the job applicants as hired and not hired. Furthermore, the DM or the nature of the problem may impose certain bounds or exact values on the number of alternatives to be assigned to a category/categories as category size restrictions. For example, a credit manager may put restriction on the number of loan applications to be accepted. Such problems are defined as constrained sorting problems (CSP) by Mousseau et al. (2003). The restrictions on the importance of the criteria can be handled by adding linear constraints to the mathematical models in function-based approaches. Category size restrictions, on the other hand, are different from the rest of the restrictions since a new measure is necessary to limit the number of alternatives to be assigned to a category.

In order to incorporate category size restrictions, Mousseau et al. (2003) define a binary variable y_{jk} in such a way that its value is one if alternative a_j is assigned to the k^{th} category and zero, otherwise. y_{jk} indicates the eligibility of the assignment of the alternatives to the categories. The study illustrates the application of the restrictions including the exact values of the category sizes in UTADIS model. The constraints in (2.11) and (2.12) ensure that the overall utility of an alternative is between the boundaries of the k^{th} category if alternative a_j is assigned by binary variable ($y_{jk} = 1$). Here, a big number, M, is included in the constraints to activate the constraints only if $y_{jk} = 1$. Let s_k be the (exact) size limit of the k^{th} category and m be the number of alternatives to be evaluated. Then, the category size restrictions are provided in (2.13). Mousseau et al. (2003) estimate a single set of parameters as in UTADIS in their application.

$$U(a_j) - u_k \leq M \cdot (1 - y_{jk}) - \varepsilon \tag{2.11}$$

$$U(a_j) - u_{k-1} + M \cdot (1 - y_{jk}) \geq 0 \tag{2.12}$$

$$\sum_{j=1}^{m} y_{jk} = s_k \quad \forall C_k \tag{2.13}$$

MIP models are also used by Özpeynirci et al. (2018) to handle category size restrictions. The study develops an interactive sorting approach for constrained sorting problems. The proposed algorithm checks for the consistency of the assignment of the alternatives with the category size restrictions. Mathematical models are solved at each iteration to diagnose and solve the inconsistencies so that minimum number of changes is required in the previous assignments of the alternatives. The authors implement their algorithm on an outranking-based method, MR-Sort and function-based sorting approach assuming a general additive preference function. The results indicate that the category size restrictions lead to elicit reasonable amount of assignment information from the DM.

2.2.1.2.3 Approaches based on probability measure

Recent studies on sorting problems have implemented the idea of defining the probability that an alternative belongs to a category. Kadzinski and Tervonen (2013) combine the ROR approach of Greco et al. (2008; 2010) with stochastic multicriteria acceptability analysis (SMAA) for sorting problems implemented with general additive preference functions. SMAA, introduced by Lahdelma et al. (1998), is a simulation technique to provide information about the weight space that represents the preferences of the DM (see Tervonen and Lahdelma (2007) for an application example). Uniform sampling of the compatible preference functions is derived by Monte Carlo simulations. The authors develop category acceptability index (CAI) for each possible category of an alternative representing the share of compatible preference functions that assign the corresponding alternative to a category. CAI is deemed as probability of belonging to a category and used to support the DM about the possible assignments of the alternatives. The randomly generated parameters to be used are rejected when they are not compatible with the assignments of the DM. The study takes attention to the high rejection rates of the sampling preference functions when the number of assignment examples increases.

Çelik et al. (2015) develop a probability measure based on L_p distance norm as a preference function. Their algorithm has a non-interactive design requiring initial assignments by the DM. Two mathematical models are solved to find the minimum and maximum utilities of alternatives as well as the category thresholds. The assignments of the DM are incorporated in the first model to minimize the classification error while maximizing the range between the maximum and minimum category thresholds as a secondary objective. The parameter values obtained in the first model are imposed to the second model to find the utility ranges of the alternatives whose categories are not known. Then, the probability of an alternative to belong to a category is determined by assuming that the minimum and maximum values have a specific (uniform or triangular) distribution function. Accordingly, probabilities are calculated and the rest of the alternatives are assigned based on these probabilities. The applications on different data sets indicate that the probabilistic approach outperforms the classification trees and UTADIS methods. The results also show that the approach may cause the misclassifications of all alternatives that belong to a category by giving no interval and thresholds to a category or categories.

Buğdacı et al. (2013) propose an interactive probabilistic approach for sorting alternatives assuming that the preferences of the DM are consistent with piecewise linear additive function. Their algorithm solves LP models to narrow down the category ranges of the alternatives. Instead of presenting the comparison between the utility of alternatives and category thresholds as in (2.9) and (2.10), they insert it to the objective function of the LP models as the difference, $U\left(a_j\right) - u_k$. The possible category of an alternative is defined based on the sign of the objective function value of the maximization and minimization models. The criterion weights and category thresholds are regarded as unknown parameters and their minimum and maximum values are estimated through additional LPs. Assuming uniform distribution among the minimum and maximum values, the authors calculate the probability that the utility of an alternative is larger than a category threshold.

If the probability that the utility of an alternative is larger than a category threshold exceeds a user-specified threshold value, $1 - \tau$, then the algorithm makes a probabilistic assignment of the alternative to the corresponding category. At the end of each iteration, an alternative is selected to be assigned by the DM. Utilizing the existing assignment information, the LP models are resolved and the probabilities are updated in the succeeding iterations. In addition to the probabilistic classification, Buğdacı et al. (2013) also consider the non-probabilistic case that correctly places alternatives to categories. The selection of the alternative to ask the DM is also based on the updated probabilities. The alternative that has the probability closest to 0.5 is regarded as the most ambiguous alternative to ask the DM. They apply their algorithm on the assignment of 81 Global MBA programs into three categories. The number of misclassified alternatives increases when τ is increased up to 0.5.

2.3 Background

We first provide the notation and then summarize the UTADIS method with its drawbacks in terms of stability and accuracy concerns. In the second part, we give the formulae, illustrate the entropy and relative entropy concepts with numerical examples, and mention the previous works on the two concepts.

2.3.1 The UTADIS method

Let $a_j = \left(a_{j1}, \ldots, a_{jn}\right)^T$ be an alternative that is evaluated with n criteria in A, the set of available alternatives, and $|A| = m$. Suppose that the alternatives are to be assigned to q predefined categories, $C_1 > C_2 > \ldots > C_q$, where C_1 and C_q are the most and least preferred categories, respectively. Let C_k be the set of alternatives that belong to the k^{th} category. UTADIS assumes an underlying additive utility function as follows:

$$U(a_j) = \sum_{i=1}^{n} u_i\left(g_i(a_j)\right) \qquad (2.14)$$

where $g_i(a_j)$ denotes the score of alternative a_j in criterion g_i. $U(a_j)$ is the overall utility of alternative a_j and $u_i\left(g_i(a_j)\right)$ denotes the marginal utility of alternative a_j

in criterion g_i. Each criterion has a marginal utility function in piecewise linear form as in Figure 5. Let g_i^{min} and g_i^{max} be the minimum and maximum scores in criterion g_i, respectively, i.e., $u_i(g_i^{min}) = 0$ and $u_i(g_i^{max}) = 1$. For each criterion g_i, the range $[g_i^{min}, g_i^{max}]$ is divided into b_i subintervals $[g_i^p, g_i^{p+1}]$ for $p = 1, \dots, b_i$.

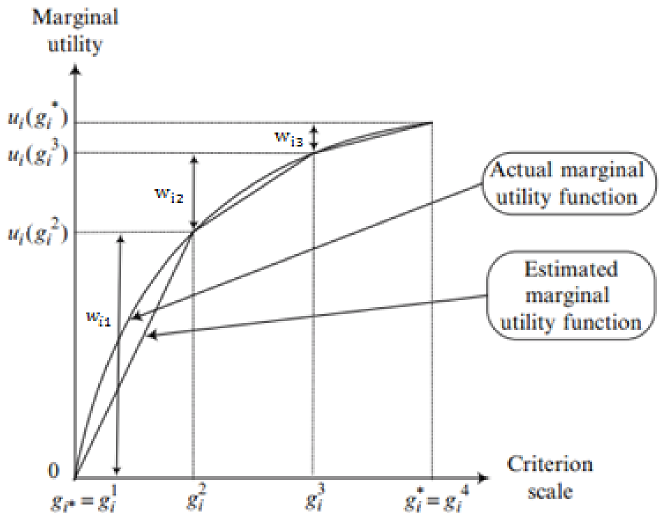


Figure 5. Piecewise linear form of marginal utility functions (Source: Doumpos and Zopounidis, 2004)

The number of subintervals, b_i can be defined by the DM or the analyst. Furthermore, two heuristic approaches are suggested to define the number of subintervals. HEUR1 (Doumpos and Zopounidis, 2002) is the basic approach to partition the subintervals so that each one contains at least one alternative. Doumpos and Zopounidis (2004) mention that HEUR1 does not consider how the alternatives are distributed among different criteria. The authors proposed HEUR2 algorithm which calculates the number of subintervals for each criterion in a way that the maximum number of subintervals at the initial step starts to decrease at each iteration through merging the subintervals as follows: A minimum threshold value for the number of alternatives in a subinterval is updated iteratively and the subintervals that include insufficient number of alternatives merge with the precedent one. Merging is performed if the number of basic variables included in the optimal solution of an iterated model is less than the number of subintervals. The goal here is to eliminate the subintervals that have redundant utility values due to redundant variables. Through extensive simulations, Doumpos and Zopounidis (2004) found that HEUR2 method improves the stability and sorting performance of the model.

Let $w_{ip} = u_i(g_i^{p+1}) - u_i(g_i^p)$ denote the utility of the subinterval p in criterion g_i. For instance, w_{i2} in Figure 5 is the difference between the utility of subinterval break points $u_i(g_i^3)$ and $u_i(g_i^2)$. The summation of the w_{ip}'s for a criterion indicates the weight of the corresponding criterion. The summation of w_{ip} values for each alternative over all subintervals and criteria is scaled to one. The piecewise marginal utility of alternative a_j can be calculated by linear interpolation for criterion g_i as follows:

$$u_i\left(g_i(a_j)\right) = \left(\sum_{p=1}^{r_{ji}-1} w_{ip} + \frac{g_i(a_j) - g_i^{r_{ji}}}{g_i^{r_{ji}+1} - g_i^{r_{ji}}} w_{ir_{ji}}\right) \tag{2.15}$$

where $g_i^{r_{ji}}$ $(1 \leq r_{ji} \leq b_i)$ denotes the score of the breakpoint that define the subinterval r_{ji}, i.e., $g_i^{r_{ji}} \leq g_i(a_j) < g_i^{r_{ji}+1}$ and $w_{ir_{ji}}$ is the utility of the corresponding subinterval r_{ji}. Finally, the overall utility of alternative a_j can be written as:

$$U(a_j) = \sum_{i=1}^{n}\left(\sum_{p=1}^{r_{ji}-1} w_{ip} + \frac{g_i(a_j) - g_i^{r_{ji}}}{g_i^{r_{ji}+1} - g_i^{r_{ji}}} w_{ir_{ji}}\right) \tag{2.16}$$

The assignment of an alternative into a category is carried out by comparing the overall utility of the alternative with the utility thresholds, u_k, of the corresponding category defined by the model as:

$$\begin{aligned}
&U(a_j) \geq u_1 \Rightarrow\ a_j \in C_1 \\
&u_k \leq U(a_j) < u_{k-1} \Rightarrow\ a_j \in C_k \quad \text{for } k = 2, \dots, q-1 \\
&U(a_j) < u_{q-1} \Rightarrow\ a_j \in C_q
\end{aligned} \tag{2.17}$$

This classification may result in some degree of misclassification. The method specifies the utility thresholds $(u_1, \dots, u_{q-1})$ and w_{ip} values in such a way that the classification error is to be minimized. For the assignment of alternative a_j into category k, the violation of the lower and upper bounds of the category are denoted by σ_j^+ and σ_j^-, respectively and defined as:

$$\begin{aligned} \sigma_j^+ &= max\{0, u_k - U(a_j)\}, \quad \forall a_j \in C_k, k \neq q \\ \sigma_j^- &= max\{0,\ U(a_j) - u_{k-1}\}, \quad \forall a_j \in C_k, k \neq 1 \end{aligned} \tag{2.18}$$

Let f_k be the number of alternatives that are assigned to category k. Then the LP for UTADIS is

$$Min \sum_{k=1}^{q} \left[\frac{\sum_{a_j \in C_k} (\sigma_j^+ + \sigma_j^-)}{f_k} \right] \tag{2.19}$$

$s.t.$

$$U(a_j) - u_1 + \sigma_j^+ \geq 0 \quad \forall a_j \in C_1 \tag{2.20}$$

$$\left.\begin{aligned} U(a_j) - u_k + \sigma_j^+ \geq 0 \\ U(a_j) - u_{k-1} - \sigma_j^- \leq \delta \end{aligned}\right\}, \quad \forall a_j \in C_k, k = 2, \dots q-1 \tag{2.21}$$

$$U(a_j) - u_{q-1} - \sigma_j^- \leq -\delta \quad \forall a_j \in C_q \tag{2.22}$$

$$u_{k-1} - u_k \geq s, \quad \forall k = 2, \dots, q-1 \tag{2.23}$$

$$\sum_{i=1}^{n} \sum_{p=1}^{b_i - 1} w_{ip} = 1 \tag{2.24}$$

$$w_{ip} \geq 0,\ \sigma_j^+, \sigma_j^- \geq 0,\ \forall i = 1, \dots, n,\ p = 1, \dots, b_i,\ j = 1, \dots, f_k \text{ for } a_j \in C_k \tag{2.25}$$

where s and δ are small positive constants. Constraints (2.20) - (2.22) define the errors for the assignment of the alternatives into categories. Constraint (2.23) enforces the threshold for a better category to be higher than that of a worse category. Constraint (2.24) scales the summation of w_{ip} values over i and p.

The LP defined by (2.19) - (2.25) estimates many parameters to find a solution. The optimal solution will generate several basic variables and so it is typical that there can be multiple optimal solutions (Köksalan and Özpeynirci, 2009). Doumpos and Zopounidis (2004) suggest conducting post optimality analysis through redesigning the LP with respect to near optimal or alternate optimal solutions. As a secondary objective, they tried to maximize $\sum_{p=1}^{b_i-1} w_{ip}$ for each criterion and u_k values for each category. To check the deviations in the accuracy and stability of the model, Doumpos and Zopounidis (2004) change (1) the number of alternatives in the reference set, (2) the number of subintervals in the criteria and (3) the classification errors in the post optimality stage. The results indicate that these differentiations significantly affect the stability and accuracy of the model.

Köksalan and Özpeynirci (2009) apply UTADIS to sort the global MBA programs assuming that the preferences of the DM are consistent with an additive function. The reference set includes 30 alternatives out of 81 alternatives. When there is no inconsistency in the preferences of the DM, the optimal solution of UTADIS will result in zero classification error and there may exist alternate optimal solutions in such a case. The authors employ eight different versions of the UTADIS model in a three-category problem of sorting the Global MBA programs. As secondary objectives, the authors tried to maximize (1) the number of alternatives in each category, (2) the weight of each criterion and (3) the category thresholds. Then, the original UTADIS model is compared with these versions with respect to average percent deviations in parameters as well as the accuracy performance of the models with respect to the misclassification rates. The results show that the classification accuracy of the non-reference alternatives is poor in most cases. Furthermore, there are significant differences in the model parameters which raises question regarding the robustness of the model parameters.

2.3.2 Entropy and relative entropy

The entropy concept was introduced by the physicist Rudolf Clausius in 1850 as a thermodynamic measure of randomness or disorder (Zhang et al., 2011). Shannon

(1948) defined it in information theory context as a measure of uncertainty of a random variable. That is, the entropy, $H(X)$, of a discrete random variable X is defined as in (2.26).

$$H(X) = -\sum_{k=1}^{q} p(x_k) log_2\, p(x_k) \qquad (2.26)$$

where the entropy is the amount of information required to find the outcome of a random variable. The logarithmic base 2 is randomly chosen in this study since the entropy values are used for comparison of the magnitudes among the alternatives and a change from base a to base b is performed through multiplication by a constant, log_b^a. Let a random variable, X, have three possible outcomes, {1,2,3}, with probabilities $1/2$, $1/4$, and $1/4$, respectively. Then, the entropy of X is calculated as

$$H(X) = -\frac{1}{2} log_2\left(\frac{1}{2}\right) - \frac{1}{4} log_2\left(\frac{1}{4}\right) - \frac{1}{4} log_2\left(\frac{1}{4}\right) = \frac{3}{2} \qquad (2.27)$$

Assume that the preference information of the DM is obtained by asking yes/no type questions. We first ask whether $X = 1$ or not since its probability is higher than the other outcomes. If the answer is no, then we ask one of the other outcomes and so we reach the outcome in either 1 or 2 questions. The expected number of questions required to determine the outcome of X is shown in (2.28).

$$Expected\ number\ of\ questions = 1 \times \frac{1}{2} + 2 \times \frac{1}{2} = \frac{3}{2} = H(X) \qquad (2.28)$$

Figure 6 shows the entropy values in case of two outcomes evaluated on log base 2. The maximum entropy is achieved when the probabilities are equal to each other which intuitively makes sense. If there is a dominant outcome with very low/high probability, then the entropy declines at an increasing rate.

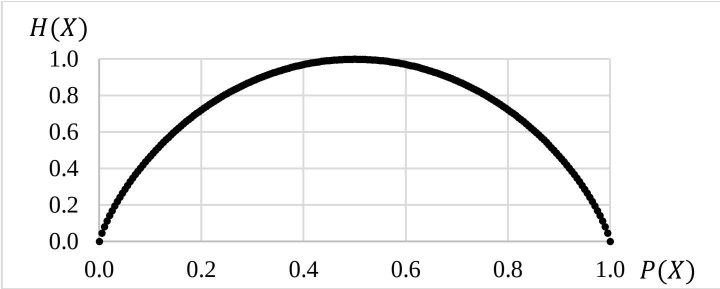


Figure 6. The entropy of a probabilistic event with two outcomes

Entropy has been widely used in studies that estimate the underlying preference function of the DM. Jaynes (1957) introduced the maximum entropy (ME) principle in case of an unknown probability distribution. According to the ME principle, the level of knowledge can precisely be identified by the largest entropy obtained from a probability distribution. Based on the ME principle, Abbas (2004) presents an adaptive question-selection algorithm for ranking problems by estimating the von Neumann and Morgenstern utility values of alternatives. Valkenhoef and Tervonen (2016) extend the work of Abbas (2004) to rank a discrete set of alternatives in which the DM is assumed to have a linear utility function. Both approaches require the DM to make pairwise comparisons between alternatives progressively. The approaches select the pair of alternatives to ask the DM that leads to the maximum reduction in the entropy of the joint distribution of the utilities which are assumed to follow a uniform distribution.

There are some approaches to the MCDM problems utilizing the concept of entropy. Xiao (2020) proposes an entropy-based fuzzy MCDM approach to model uncertainty and rank the alternatives. Karakaya et al. (2016) develop a multi-objective feature selection approach and measure the relation between two variables based on entropy. In another approach, Ciomek et al. (2017) estimate the reduction in entropy by selecting pairwise questions to rank the alternatives. The authors utilize the rank acceptability index (RAI) estimated by Monte Carlo simulations. RAI is defined as the share of utility functions that lead to a specific rank for an alternative. The entropy is calculated based on the RAI scores. Wu et al. (2011) use entropy to find the cross-efficiency scores of the alternatives as an alternative method for data envelopment analysis.

When the number of outcomes vary, the entropy value may fail to capture uncertainty levels. For instance, suppose that there are two alternatives to be assigned into any of three categories. Let the possible categories that alternatives 1 and 2 can be assigned to be $\{C_1, C_2\}$ and $\{C_1, C_2, C_3\}$, respectively with the corresponding probabilities of $\{0.50, 0.50\}$ and $\{0.15, 0.15, 0.70\}$. The entropies of alternatives 1 and 2 are calculated as 1 and 1.18, respectively. Although it is expected for alternative 1 to have higher uncertainty, its entropy level is lower due to different number of possible categories to be placed between the alternatives. Relative entropy, $H_R(X)$, on the other hand, considers the number of possible categories by dividing the entropy of an alternative to the maximum entropy with the same number of possible categories (Shannon, 1948). The relative entropies of alternatives 1 and 2 are calculated as 1.00 and 0.75, respectively by using the formula in (2.29). In Figure 7, an illustration of entropy and relative entropy values for 50 alternatives with different probability values in three-outcome case is shown. The maximum relative entropy value is one while the maximum entropy value is $log_2 3 = 1.585$.

$$H_R(X) = \frac{-\sum_{k=1}^{q} p(x_k) log_2 \, p(x_k)}{log_2 \, q} \tag{2.29}$$

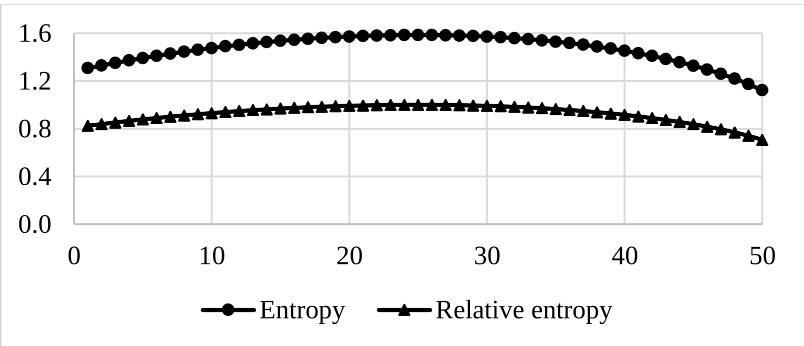


Figure 7. Entropy and relative entropy example

CHAPTER 3

INTERACTIVE APPROACHES FOR SORTING PROBLEMS

We present the proposed interactive approaches for sorting the alternatives evaluated on multiple criteria. We first define the problem environment and present mathematical models for the preferences of the DM represented by (i) piecewise linear and (ii) monotone non-decreasing additive preference functions in Sections 3.1 and 3.2, respectively. In addition to the LP models in each function, we also present the MIP models for the case of category size restrictions. We then provide the steps of the algorithms for the non-probabilistic and probabilistic cases in Sections 3.3 and 3.4, respectively as well as the benchmark algorithms in each case.

We modified the LP models developed by Buğdacı et al. (2013) and use their notation in order to facilitate easy reading. Recall that set A consists of m alternatives $a_1, a_2, \dots, a_m$ that are evaluated on n criteria and the DM is required to assign the alternatives into q categories $C_1, C_2, \dots, C_q$ where C_1 and C_q are the most and least preferred categories, respectively. Recall that C_k is the set of alternatives that belong to the k^{th} category. Let C_0 be the set of alternatives categories of which are not known.

3.1 Piecewise linear additive preference functions

In this section, we assume that the preferences of the DM are consistent with an additive utility function where the marginal utilities of alternatives are piecewise linear in each criterion as in UTADIS. $LP1_{a_t,k}$ and $LP2_{a_t,k}$ are the two models that are used to define the category ranges of the alternatives for piecewise linear form when no category size restriction is addressed in the problem. We note that rather than finding a single set of parameters as in UTADIS, the whole set of parameters is considered to avoid any misclassification.

Model $(LP1_{a_t,k})$

$$Min\ U(a_t) - u_k \tag{3.1}$$

$s.t.$

$$U(a_j) = \sum_{i=1}^{n}\left(\sum_{p=1}^{r_{ji}-1} w_{ip} + \frac{g_i(a_j) - g_i^{r_{ji}}}{g_i^{r_{ji}+1} - g_i^{r_{ji}}} w_{ir_{ji}}\right), \forall a_j \in A \tag{3.2}$$

$$U(a_j) \geq u_k,\ \forall a_j \in C_k, \quad k = 1, \dots, q-1 \tag{3.3}$$

$$U(a_j) \leq u_{k-1} - \delta,\ \forall a_j \in C_k, \quad k = 2 \dots, q \tag{3.4}$$

$$u_{k-1} - u_k \geq \delta, \quad k = 2, \dots, q-1 \tag{3.5}$$

$$u_{q-1} \geq \delta \tag{3.6}$$

$$u_1 \leq 1 - \delta \tag{3.7}$$

$$\sum_{i=1}^{n}\sum_{p=1}^{b_i-1} w_{ip} = 1 \tag{3.8}$$

$$w_{ip} \geq 0, \quad i = 1, \dots, n,\ p = 1, \dots, b_i \tag{3.9}$$

where δ is a small positive constant that is set as 10^{-3}. The objective of $(LP1_{a_t,k})$ is to minimize the difference between the utility of alternative a_t and the threshold of the k^{th} category. Constraint (3.2) determines the utility of each alternative based on a piecewise linear additive function. Constraint sets (3.3) and (3.4) ensure that the alternatives assigned by the DM are within the boundaries of their exact categories. Constraint (3.5) guarantees that the utility threshold of a better category is higher than that of a worse category. In constraints (3.6) and (3.7), the lower and upper limits of the category thresholds are set in a way that the most and least preferred categories have utility intervals. Constraint (3.8) stands for normalization of the w_{ip} values and constraint (3.9) is the nonnegativity constraint. In the second model, $(LP2_{a_t,k})$, we retain all constraints from (3.2) to (3.9) and change the objective function to maximize the difference between the utility of alternative a_t and the utility threshold of the k^{th} category.

Model $\left(LP2_{a_t,k}\right)$

$$Max\ U(a_t) - u_k \tag{3.10}$$

$s.t. \quad (3.2) - (3.9)$

In order to reflect the category size restrictions in the models, the numbers of alternatives in categories are reflected in the mathematical models. To be able to incorporate such restrictions, a binary variable y_{jk} is defined in such a way that its value is one if an alternative a_j is assigned to the k^{th} category and zero, otherwise as in Mousseau et al. (2003). Then the problem can be solved by the following MIP models:

Model $\left(MIP1_{a_t,k}\right)$

Min $U(a_t) - u_k$

$s.t. \quad (3.2) - (3.9),$

$$U(a_j) \geq u_k - M(1 - y_{jk}),\ \forall a_j \in C_0, \quad k = 1, \dots, q-1 \tag{3.11}$$

$$U(a_j) \leq u_{k-1} + M(1 - y_{jk}) - \varepsilon,\ \forall a_j \in C_0, \quad k = 2 \dots, q \tag{3.12}$$

$$\sum_{k=1}^{q} y_{jk} = 1 \quad \forall a_j \in A \tag{3.13}$$

$$\sum_{j=1}^{m} y_{jk} = s_k \quad \forall C_k \in C^S \tag{3.14}$$

$$y_{jk} \in \{0,1\}, \qquad \forall j,k \tag{3.15}$$

where M is a large positive constant, $s_k = |C_k|$ is the given size of the k^{th} category and C^S is the set of categories exact category sizes of which are known. Constraints (3.11) and (3.12) ensure that the utility of unlabeled alternatives are within the boundaries of a category if an alternative is assigned to the category through binary variable y_{jk}. Constraint (3.13) states that each alternative can be assigned to a single category. Category size restrictions are handled in constraint (3.14). Our models are flexible so that other types of category size constraints such as bounds or comparisons can be easily added. In the second model, we maximize the difference between the utility of alternative a_t and the utility threshold of the k^{th}category through keeping all constraints from (3.2) to (3.9) and (3.11) to (3.15).

Model $\left(MIP2_{a_t,k}\right)$

Max $U(a_t) - u_k$

$s.t.$ $\quad (3.2) - (3.9), (3.11) - (3.15)$

3.2 General monotone additive preference functions

When the preferences of the DM are assumed to be consistent with a general additive utility function in monotonically non-decreasing form, then there is neither characteristic point nor subinterval that are included in piecewise linear utility functions as in UTADIS. For general additive utility functions, the criteria scores are ordered from least preferred to most preferred ones. Let $x_1^i, x_2^i, \ldots, x_{m_i}^i$ be the ordered score values in criterion g_i where $x_h^i < x_{h+1}^i$, $h=1, \ldots, m_i - 1$, $m_i \leq m$. Recall that the marginal utility of alternative a_j in criterion g_i is defined as $u_i\left(g_i(a_j)\right)$ where the overall utility of a_j, $U(a_j)$, is the sum of the marginal utilities in each criterion.

Model $\left(LP3_{a_t,k}\right)$

$Min\ U(a_t) - u_k$

$s.t.$ $\quad (3.3) - (3.7),$

$$u_i(x_h^i) \leq u_i(x_{h+1}^i), h = 1, \ldots, m_i - 1, \qquad i = 1, \ldots, n \tag{3.16}$$

$$u_i(x_1^i) = 0, \ \ i = 1, \ldots, n \tag{3.17}$$

$$\sum_{i=1}^{n} u_i(x_{m_i}^i) = 1 \tag{3.18}$$

$$U(a_j) = \ \sum_{i=1}^{n} u_i\left(g_i(a_j)\right), \ \forall a_j \in A \tag{3.19}$$

Constraints (3.16) - (3.18) define the marginal and overall utilities of each alternative based on an additive utility function. Constraint (3.16) guarantees that smaller criteria scores have lower marginal utilities in each criterion. Constraints (3.17) and (3.18) stand for normalization, i.e., ensure that the overall utilities are within the range of $[0,1]$. The overall utility of an alternative is defined as the summation of the marginal utilities in each criterion in Constraint (3.19).

In Model $\left(LP4_{a_t,k}\right)$, we maximize the difference between the utility of alternative a_t and the utility threshold of category k including all constraints from (3.3) to (3.7) and (3.16) to (3.19).

Model $\left(LP4_{a_t,k}\right)$
$Max\ U(a_t) - u_k$
$s.t.$ (3.3) − (3.7), (3.16) to (3.19)

In order to incorporate the category size restrictions in general additive preference functions, we add the binary constraints to $(LP3_{a_t,k})$ and $(LP4_{a_t,k})$ and the resulting MIP models are as follows:

Model $\left(MIP3_{a_t,k}\right)$
Min $U(a_t) - u_k$
$s.t.$ (3.3) − (3.7), (3.11) − (3.19)

Model $\left(MIP4_{a_t,k}\right)$
Max $U(a_t) - u_k$
$s.t.$ (3.3) − (3.7), (3.11) − (3.19)

3.3 The approach for non-probabilistic case

In this section, we present the proposed approach when the probabilistic assignments are not allowed. We first explain how the mathematical models and the dominance relations are used to identify the possible category ranges of the alternatives. We then provide the steps of the algorithm for non-probabilistic case. We note that our algorithm is explained in terms of the LP models of unconstrained sorting problem.

The algorithm for constrained problem is similar to the algorithm for unconstrained problem except that the mathematical models are MIPs instead of LPs.

3.3.1 The non-probabilistic algorithm

At each iteration of the algorithm, we solve LP models for each unlabeled alternative to narrow down the possible category ranges of the alternatives. The LP models search for the compatible preference functions to assign the alternatives into categories. If there is no such set of parameters for a category, then this category is eliminated from the possible category range of the alternatives. Once the models narrow down the category ranges, we select an alternative to be asked the DM by the selection method explained in Section 3.3.2. We feed the models with the category information of the alternatives identified by the DM in each iteration until all alternatives are assigned to their true categories.

The category ranges of the alternatives have no jumps, i.e., if C_k and C_{k+2} are among the possible categories of an alternative, then this alternative can also be assigned to C_{k+1} (Greco et al., 2010). Hence, we consider the assignment eligibility of the alternatives in a sequential order of the possible categories. Since $LP1_{a_t,k}$ and $LP3_{a_t,k}$ are minimization models, a nonnegative objective function value indicates that there are no w_{ip} and u_k values that make the category threshold larger than the utility of the alternative. Hence, the worst possible category of alternative a_t is C_k. Note that if the objective function value in the minimization problem is nonnegative for utility threshold of the k^{th} category, then it will be nonnegative for the thresholds of worse categories, i.e., $k+1$ to q. Therefore, there is no need to search for the worse categories once we obtain a nonnegative value in the minimization problem. It is efficient to start from the best possible category when we search for the worst category of an alternative through minimization problem.

$LP2_{a_t,k}$ and $LP4_{a_t,k}$ are maximization models and a negative objective function value shows that no w_{ip} and u_k values can make the utility of the alternative larger than the category threshold. Hence, such an alternative cannot be in C_k and the best possible category of alternative a_t is C_{k+1}. If the objective function value in the maximization problem is negative for a category threshold u_k, then it will be negative for larger thresholds as well. Thus, it is efficient to start from the worst possible category when we search for the best category of an alternative through maximization problem. If the objective function value in the minimization problem is nonnegative, then it is also nonnegative for maximization problem. This suggests to skip the maximization problem whenever a nonnegative value is obtained in the minimization problem. Hence, we start with the minimization problem to identify the category ranges.

Definition 1. Let a_x and a_y be two distinct alternatives, $a_x, a_y \in A$. Alternative a_y dominates alternative a_x if and only if a_y is at least as good as a_x in all criteria and better in at least one criterion.

Remark 1. Let a_x and a_y be two alternatives where alternative a_y dominates alternative a_x. If the worst possible category of alternative a_x is $C_{k'}$, then the worst category of a_y is $C_{k'}$.

Remark 2. Let a_x and a_y be two alternatives where alternative a_x dominates alternative a_y. If the best possible category of alternative a_x is $C_{k'}$, then the best category of a_y is $C_{k'}$.

According to Remarks 1 and 2, the dominance relations between the alternatives can be used to make further category range reductions. As in Ulu and Köksalan (2001; 2014) we utilize the dominance relations to decrease the total number of models solved to complete the task. We check the dominance relations at the beginning of the decision process for once and update the category ranges whenever the category range of an alternative is narrowed down by LPs or after we ask the DM to make an assignment.

We next provide the steps of the non-probabilistic algorithm, A_{RENT}. We note that we present and describe the algorithm for an additive utility function in piecewise linear form in the rest of this section. $(LP1_{a_t,k})$ and $(LP2_{a_t,k})$ will be replaced with $(LP3_{a_t,k})$ and $(LP4_{a_t,k})$ in case of monotonically non-decreasing additive utility function.

Recall that A is the set of available alternatives, C_0 is the set of alternatives whose categories are not known and C_k is the set of alternatives in the k^{th} category. Let C_t^W and C_t^B be the worst and best categories of alternative a_t, respectively. We assume that at the beginning there is no information about the categories of the alternatives, i.e., $C_0 = A$.

3.3.1.1 Algorithm A_{RENT}

Step 0: Set $C_0 = A$, $C_k = \emptyset$ for $k = 1, \dots, q$, $C_t^W = q$ and $C_t^B = 1$ for each $a_t \in C_0$.

Step 1: For each $a_t \in C_0$, set $k = C_t^B$.

1.1. Solve $(LP1_{a_t,k})$.

(i) If $obj_1^*(a_t, k) \geq 0$, set $C_t^W = k$. Set $C_{t'}^W = k$ for each alternative $a_{t'} \in C_0$ that dominates alternative a_t and if $C_{t'}^W = C_{t'}^B = k$, let $C_k \leftarrow C_k \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_t^W = C_t^B = k$, then let $C_k \leftarrow C_k \cup \{a_t\}$, $C_0 \leftarrow C_0 - \{a_t\}$ and go to Step 1.3. Otherwise set $k = k - 1$ and go to Step 1.2.

(ii) If $obj_1^*(a_t, k) < 0$ and $k < C_t^W - 1$, set $k = k + 1$ and repeat Step 1.1.

(iii) If $obj_1^*(a_t, k) < 0$ and $k = C_t^W - 1$, go to Step 1.2.

1.2. Solve $(LP2_{a_t,k})$.

(i) If $obj_2^*(a_t, k) < 0$, set $C_t^B = k + 1$. Set $C_{t'}^B = k + 1$ for each alternative $a_{t'} \in C_0$ that is dominated by alternative a_t and if $C_{t'}^W = C_{t'}^B = k + 1$, then let $C_{k+1} \leftarrow C_{k+1} \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_t^W = C_t^B = k + 1$, then let $C_{k+1} \leftarrow C_{k+1} \cup \{a_t\}$ and $C_0 \leftarrow C_0 - \{a_t\}$. Go to Step 1.3.

(ii) If $obj_2^*(a_t, k) \geq 0$ and $k > C_t^B$, set $k = k - 1$ and repeat Step 1.2.

(iii) If $obj_2^*(a_t, k) \geq 0$ and $k = C_t^B$, go to Step 1.3.

1.3. If the category range of all alternatives are reduced, then go to Step 2. Otherwise go to step 1.1 for the next $a_t \in C_0$.

Step 2: Select an alternative $a_t^s \in C_0$ to ask the DM. The DM assigns alternative a_t^s to $C_{k'}$, then let $C_{k'} \leftarrow C_{k'} \cup \{a_t^s\}$ and $C_0 \leftarrow C_0 - \{a_t^s\}$. Set $C_{t'}^W = k'$ for each alternative $a_{t'} \in C_0$ that dominates alternative a_t^s. Set $C_{t'}^B = k'$ for each alternative $a_{t'} \in C_0$ that is dominated by alternative a_t^s. If $C_{t'}^W = C_{t'}^B$, let $C_{k'} \leftarrow C_{k'} \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_0 = \emptyset$, go to step 3. Otherwise go to Step 1.

Step 3: Present the categories of alternatives to the DM and stop.

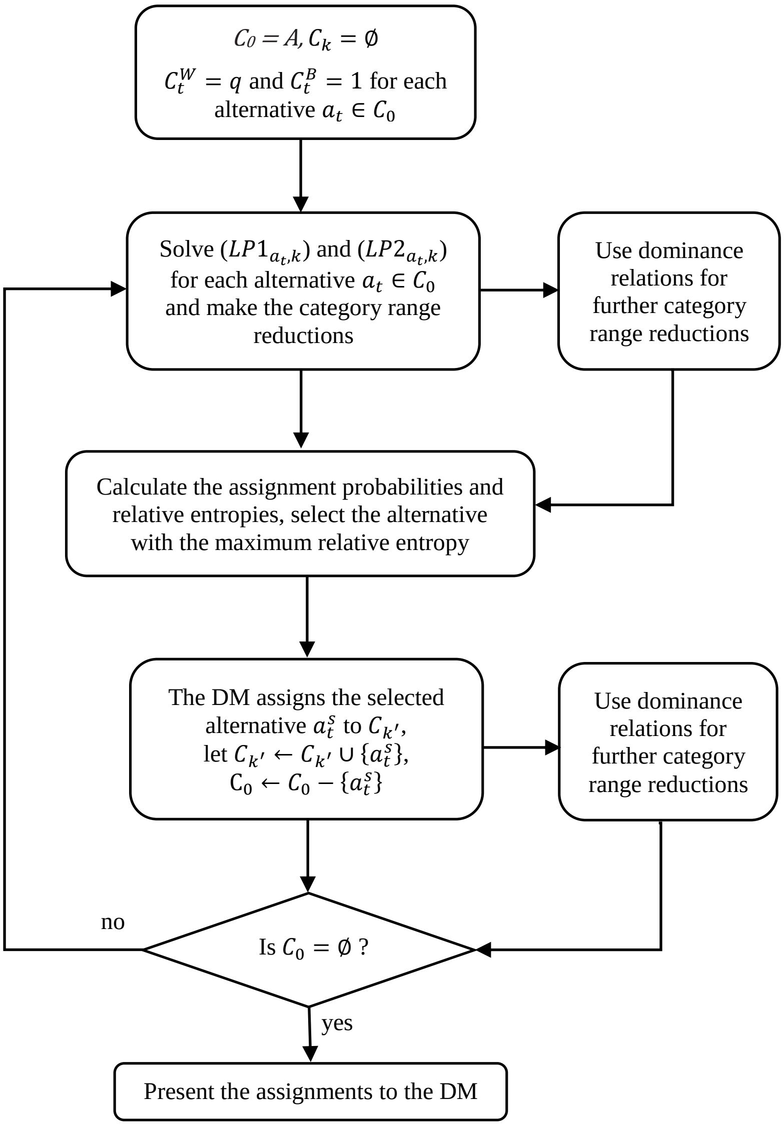


Figure 8. Flowchart of A_{RENT}

Our non-probabilistic algorithm, A_{RENT}, does not require an initial reference set established by the DM. Instead, we consult the DM iteratively. In Step 1 of the algorithm, we define the worst and best possible categories by solving $LP1_{a_t,k}$ and $LP2_{a_t,k}$ for each alternative whose category has not been defined yet; considering all possible category thresholds. We also utilize the dominance relations for further category range reductions. In Step 2, regardless of whether an alternative is assigned to a category or not at the end of Step 1, the algorithm selects an alternative to ask the DM. Once the DM specifies the category of the selected alternative, we update the category information of this alternative as well as the ones that have a dominance relation with this alternative. We keep executing the steps until all alternatives are assigned to categories. Figure 2 presents the flowchart of the algorithm.

3.3.2 Selection of the alternative to ask the DM

Regardless of the type of the decision problem, gathering information from the DM is an important step in the modelling process. In interactive approaches, the selection of the questions affects the amount of information required to complete the task (Holloway and White, 2003; Branke et al., 2017). The aim here is to derive as much new information as possible about the preferences of the DM. Receiving the category information of the alternatives that have high assignment ambiguity or uncertainty may lead the model parameters to converge to their true values faster. Hence, an uncertainty measure may be utilized to select the alternative to be asked the DM. The assignment probabilities can provide valuable information regarding the assignment uncertainties of alternatives. If an alternative has similar probability values for the categories, then this alternative can be regarded as ambiguous since it is equally probable that this alternative can be assigned to each category.

The previous probabilistic approaches on sorting problems usually assume that the decision parameters follow a specific probability distribution such as uniform or Gaussian distribution. The assignment probabilities are defined by the parameters of mathematical models in Buğdacı et al. (2013) and Çelik et al. (2015). These two studies assume that the parameters follow specific distributions among their minimum and maximum values. Mathematical models are solved to find the boundaries of the decision parameters. By this way, the assignment probabilities are estimated to probabilistically assign the alternatives. Buğdacı et al. (2013) further utilize the assignment probabilities to select the alternative to be asked the DM.

The second way of defining the assignment probabilities is to gather information through several hypothetical assignments of the alternatives. Tervonen et al. (2009) and Kadzinski and Tervonen (2013) hypothetically assign the alternatives based on the parameters obtained from several Monte Carlo simulations. The random samples of decision parameters are generated assuming uniform distribution. The assignment frequency of an alternative in a category is transformed to the probability measure called CAI. CAI of a category is defined as the share of hypothetical assignments that assign an alternative to a category. That is, CAI represents the probability of belonging to a category. One drawback of this method is that the randomly generated samples of parameters are rejected to be used when they are not compatible with the assignments of the DM. We introduce our model-based and simulation-based hypothetical assignment approaches in the next sections.

3.3.2.1 Model-based hypothetical assignment approach

In simulation-based hypothetical assignment approach, the rejection rates due to the incompatibility can cause extensive time to generate sufficient number of random samples (Tervonen et al., 2013). Instead, we utilize a set of compatible preference functions obtained from mathematical models that are good estimators of the category thresholds and utilities of the alternatives to make hypothetically assignments.

We develop an ad hoc procedure to hypothetically assign the alternatives. Recall that m is the number of alternatives and $q-1$ is the number of category thresholds where the number of categories is equal to q. In the first iteration of the algorithm, we solve $2 \cdot m \cdot (q-1)$ LP models and the number of models decreases in the following iterations either with category range reductions or assignments of the DM. $(LP1_{a_t,k})$ and $(LP2_{a_t,k})$ find different $U(a_t)$ and u_k values which are expected to converge to their true values as the algorithm progresses. When ($LP1_{a_t,k}$) and ($LP2_{a_t,k}$) are solved to consider the possible assignment of alternative a_t to C_k, we find the minimum and maximum values for the category threshold, u_k. Throughout the iterations the category thresholds are expected to converge to the average of their minimum and maximum values. Hence, we take the average of u_k values found by ($LP1_{a_t,k}$) and ($LP2_{a_t,k}$) to define the thresholds that separate the categories. We also take the average of $U(a_t)$ values found by the two models. We then make a hypothetical assignment with the average values for each unlabeled alternative. Thus, each alternative is hypothetically assigned to a category whenever the two models are solved to identify the possible assignment of an alternative.

Lemma 1. The assignments with average $U(a_t)$ and u_k values are compatible with the assignments of the DM.

Proof. Let alternative a_{DM} is an assignment of DM to category k', $C_{k'}$. When ($LP1_{a_t,k}$) and ($LP2_{a_t,k}$) are solved, the utility of alternative a_{DM} will be between the thresholds derived from the two models as in (3.20) and (3.21). Then, the average utility of alternative a_{DM} will be between the average of the upper and lower thresholds as in (3.22). □

$$u_{k'-1}^{LP1} \leq U(a_{DM}^{LP1}) < u_{k'}^{LP1} \tag{3.20}$$

$$u_{k'-1}^{LP2} \leq U(a_{DM}^{LP2}) < u_{k'}^{LP2} \tag{3.21}$$

$$\frac{u_{k'-1}^{LP1} + u_{k'-1}^{LP2}}{2} \leq \frac{U(a_{DM}^{LP1}) + U(a_{DM}^{LP2})}{2} < \frac{u_{k'}^{LP1} + u_{k'}^{LP2}}{2} \tag{3.22}$$

Lemma 1 indicates that a compatible preference function can be derived by using the average values of the two models. At the end of each iteration, we sum the number of assignments in all the models solved. For instance, $m \cdot (q-1)$ hypothetical assignments are considered in the first iteration assuming that there is no initial assignment obtained from the DM.

3.3.2.2 Simulation-based hypothetical assignment approach

In addition to the model-based hypothetical assignment approach, we use a simulation-based hypothetical assignment approach to find the assignment frequency of the alternatives. Tervonen et al. (2009) hypothetically assign alternatives by generating 10,000 sets of parameters (profiles, thresholds and weights in ELECTRE-TRI) using Monte Carlo simulation technique. Previous studies claim that 10,000 Monte Carlo iterations is sufficient to achieve 95% confidence (Milton and Arnold, 1995; Tervonen and Lahdelma, 2007). Kadzinski and Tervonen (2013) hypothetically assign the alternatives by Monte Carlo simulation assuming a general monotone preference function for the DM. Recall that $x_1^i, x_2^i, \dots, x_{m_i}^i$ are the ordered score values in criterion g_i in general monotone preference function case. Hence, the marginal utility of m_i criteria scores are generated for each criterion g_i. The authors applied their approach in a 5-criteria, 4-category and 27-alternative problem by requiring the DM to initially assign nine alternatives to categories.

Recall that the randomly generated parameters may not be compatible with the assignments of the DM. For instance, the estimated utility of an alternative belonging to a more preferred category may be lower than that of an alternative that belongs to a worse category. In such a case, the randomly generated set of parameters is rejected. The rejection rates due to the incompatibility are expected to increase at an increasing rate with the addition of new assignments of the DM. The authors declare that the rejection sampling with the category information of nine assigned alternatives takes approximately 20 seconds in a single iteration. In this study, preliminary experiments in 8-criteria, 4-category and 76-alternative problem assuming a general monotone preference function show that it takes hours to generate 10,000 compatible sets of parameters when the DM assigns 10 alternatives to the categories. Hence, we do not consider the general monotone preference functions in simulation-based hypothetical assignment approach.

In order to apply the simulation-based approach, the preferences of the DM are assumed to be consistent with a piecewise linear function. For hypothetical assignments, we generate 10,000 random sample sets with Monte Carlo simulations assuming uniform distributions for parameters. We introduce an efficient approach to generate 10,000 compatible sets of parameters. In piecewise linear utility function case involving n criteria and b_i subintervals for each criterion g_i, total number of w_{ip} values generated in each sample is $\sum_{i=1}^{n} b_i$ and the summation of the w_{ip} values is equal to one as in (3.8). Rubinstein (1982) presents two random weight generation techniques when the summation of parameters is equal to one as

Method-1: Rubinstein-1

Step 1: *Generate n random variates r_i from U(0, 1).*

Step 2: $Tot = \sum_{i=1}^{n} r_i$.

Step 3: $w_i = \frac{r_i}{Tot}$.

Method-2: Rubinstein-2

Step 1: *Generate $n-1$ random variates r_i from U(0, 1).*

Step 2: *Sort r_i's in increasing order:* $r_1, r_2, \ldots, r_{n-1}$.

Step 3: $w_1 = r_1, w_2 = r_2 - r_1, \ldots, w_{n-1} = r_{n-1} - r_{n-2}, w_n = 1 - r_{n-1}$.

Tervonen et al. (2009) and Kadzinski and Tervonen (2013) use *Rubinstein-2* to generate uniformly distributed random numbers between zero and one. Kim et al. (2006) perform intensive experiments to compare the two techniques. Although the CPU time of random number generation by *Rubinstein-2* is less than that of *Rubinstein-1*, the difference between the CPU times of the two techniques is found to be statistically insignificant. We modify *Rubinstein-1* to generate the w_{ip} values. The ranges of w_{ip} values are expected to narrow down with the addition of new assignments by the DM. The DM may also provide ranges for these values as in constrained sorting problems. If that is not the case, then $LP5_{i,p}$ and $LP6_{i,p}$ are solved for each criterion g_i and subinterval *p* to find the range of w_{ip} values in each iteration. We then generate w_{ip} values from their ranges assuming that they are uniformly distributed within these ranges. Afterwards the w_{ip} values are scaled to ensure that their summation over *p* is equal to one for each i as in *Rubinstein-1*. Although the scaled weight values may not fall within the acceptable ranges, drawing the w_{ip} values from their ranges is expected to result in higher compatibility with the preferences and lower rejection rates.

Model $\left(LP5_{i,p}\right)$

$$Min\ w_{ip} \tag{3.23}$$

$s.t.\quad (3.2) - (3.9)$

Model $\left(LP6_{i,p}\right)$

$$Max\ w_{ip} \tag{3.24}$$

$s.t.\quad (3.2) - (3.9)$

After calculating the utilities of alternatives based on the generated parameter values, u_k values are generated to define the category thresholds. In the first few iterations, there may be categories without any alternatives assigned. In such a case, we generate the category thresholds by uniform distribution within the utility interval of the alternatives. When there is at least one alternative assigned by the DM or the mathematical models in each of the consecutive categories C_k and C_{k+1}, u_k value is generated from uniform distribution in the range between the utility of alternative with the highest value in C_{k+1} and the utility of alternative with the lowest value in C_k.

Lemma 2. If alternative a_t is assigned to C_k at least one time, then C_k is one of the possible categories of alternative a_t.

Proof. The assignment of alternative a_t to C_k indicates that there is at least one set of parameters compatible with the preferences of the DM that can assign alternative a_t to C_k. Since the mathematical models search for compatible sets of parameters, then C_k cannot be eliminated from the possible category range of alternative a_t. □

Based on the estimated utilities and category thresholds, alternatives are assigned to the categories 10,000 times in each iteration. Lemma 2 explains the relationship between the assignments and the possible category range of the alternatives defined by the mathematical models. According to Lemma 2, it is efficient to make the assignments before solving the mathematical models. If an alternative is not assigned to one of its possible categories, then we solve the corresponding model to make sure that this alternative cannot be assigned to that category. By this way, we expect to minimize the number of models solved to complete the decision task.

3.3.2.3 Relative entropy selection method

Let x_{tk} be the number of assignments of alternative a_t in category k at an iteration. The frequency of an alternative belonging to a category is converted to a probabilistic measure as

$$p(y_{tk} = 1) = \frac{x_{tk}}{\sum_{r=1}^{q} x_{tr}} \qquad (3.25)$$

where y_{tk} takes the value of one if alternative a_t belongs to category k and zero, otherwise.

The assignment probabilities can give an idea about the potential category/categories of the alternatives. That is, if the probability of belonging to a category is much higher than those of other categories for an alternative, then it can be inferred that the alternative is much more likely to belong to this category. Hence, it may not be wise to ask the DM to place such alternatives. The aim here is to derive as much new information as possible about the preferences of the DM. Asking the most ambiguous one may lead the model parameters to converge to their true values faster. Whenever an alternative has similar probability values for each category, then this alternative is selected to be asked the DM for placement. Therefore, we need to quantify the uncertainty level of the alternatives in order to find the most ambiguous one.

The entropy concept fits well to our measurement of uncertainty about the categories of the alternatives. It is expected to obtain more valuable information by the assignment of the alternatives having high uncertainty levels about the possible categories to be placed. As stated before, probabilistic events that have different number of outcomes can be compared with respect to their relative entropy levels. In this study, we select the alternative with the maximum relative entropy to ask the DM for assignment as in (3.26). To break the ties in case of equal relative entropy for multiple alternatives, we select the one that has highest number of dominance relations with the alternatives that have not been assigned yet as in Özpeynirci et al. (2018).

$$a_t^s = \arg\max_{a_j \in C_0} H_R(a_j) = \arg\max_{a_j \in C_0} \frac{-\sum_{k=1}^{q} p(y_{jk} = 1) log_2\, p(y_{jk} = 1)}{log_2\, q} \qquad (3.26)$$

3.3.3 Benchmark algorithms for non-probabilistic case

In order to make a fair comparison, we consider threshold-based sorting algorithms for additive preference functions as benchmark algorithms where the assignment information from the DM is gathered progressively. We first explain the algorithm of Buğdacı et al. (2013) as one of the benchmark algorithms. Then we define three alternative selection techniques that are employed in our algorithm. In these benchmark algorithms, different methods are used to select the alternative to ask DM.

3.3.3.1 Algorithm of Buğdacı et al. (2013)

The first benchmark algorithm is the non-probabilistic algorithm of Buğdacı et al. (2013), $A_{BU\check{G}}$. The authors construct LP models to calculate the probability that the utility of an alternative is larger than the utility threshold of a category. The alternative that has the probability closest to 0.5 is regarded as the most ambiguous alternative to ask the DM as in (3.27).

$$a_t^s = \arg\min_{a_j \in C_0} \min_k \left| p(U(a_j) > u_k) - 0.5 \right| \qquad (3.27)$$

If the probability is equal to 0.5, then it is perceived as equally likely to have a utility of the selected alternative that is greater or smaller than the threshold of a category.

We convert the mathematical models of Buğdacı et al. (2013) into the MIPs in case of category size restrictions.

3.3.3.2 Alternative selection approaches

In MCS problems, there are several approaches for the selection of the alternatives to be assigned by the DM. Assuming a quasiconcave preference function, Ulu and Köksalan (2014) solve LP models to find the middle most weights that can represent the preferences of DM. Then they calculate the aggregate utilities of the alternatives based on the middle most weights derived from the LP models. The bounds of the categories are determined by the minimum and maximum utilities of the alternatives that are already assigned by the DM. Hence, they propose assignment-based sorting approach rather than a threshold-based one. If there exists an alternative having aggregate utility outside the boundaries of the categories, then this alternative is selected to be asked the DM. In case that there is no such alternative, then the algorithm selects the alternative that has an aggregate utility closest to the boundary of a category. These alternatives are regarded as the ambiguous ones in terms of assignment tendency.

We include three alternative selection techniques that are employed in our algorithm. Firstly, we randomly select an alternative among the unlabeled ones to be assigned by the DM. Since there is randomization in the selection process, we randomly generate 100 samples for each problem setting and report the averages. Secondly, we use the idea of Özpeynirci et al. (2018) where the alternative to be selected to ask the DM is identified by considering the alternative(s) with the highest cardinality of set of possible categories. The authors argue that the wider the category range of an alternative is, the more information is obtained. If there are multiple alternatives with the highest cardinality, then the one with the highest number of dominance relations with the rest of the unlabeled alternatives is selected. Dominance relations are expected to provide additional improvements in the assignments of the alternatives that are not assigned.

The third selection method is the maximal minimax regret approach of Benabbou et al. (2017). The minimax regret approach, proposed by Savage (1951), is usually used to minimize the potential loss in investments that may arise when the worst-case scenario occurs in the financial markets. The maximal regret, $MR(a_j, C_k)$, occurs due to the incorrect assignments of the alternatives and is measured by the maximum difference between thresholds and the utilities of the alternatives as in (3.28) where δ is a small positive constant. The mathematical models, $(LP1_{a_t,k})$ and $(LP2_{a_t,k})$ give the maximum difference between the utilities of the alternatives and category thresholds. For each alternative, the minimum $MR(a_j, C_k)$ value among the possible categories is referred to as the minimax regret. The alternative with maximal minimax regret is selected to be assigned by the DM as in (3.29). In case of equal maximal minimax regret among the alternatives, an alternative with higher number of dominance relations with the other unlabeled alternatives is selected as in Özpeynirci et al. (2018).

$$MR(a_j, C_k) = \max_{U(a_j), u_k} \max\{u_k - U(a_j), U(a_j) - u_{k-1} + \delta, 0\} \tag{3.28}$$

$$a_t^s = \arg\max_{a_j \in C_0} \min_{C_k \in [C_t^W, C_t^B]} MR(a_j, C_k) \tag{3.29}$$

The performances of our algorithms with relative entropy selection based on model-based hypothetical assignments (A_{RENTM}) and simulation-based hypothetical assignments (A_{RENTS}) are compared against the non-probabilistic algorithm of Buğdacı et al. (2013) ($A_{BU\check{G}}$), the random selection (A_{RAND}), the selection method of Özpeynirci et al. (2018) ($A_{\ddot{O}ZP}$) as well as the maximal minimax regret selection approach of Benabbou et al. (2017) (A_{REG}) through experiments for non-probabilistic case in Chapter 4.

3.4 The approach for probabilistic case

The assignment probabilities can give an idea about the potential category/categories of the alternatives. That is, if the probability of belonging to a category is much higher than those of other categories for an alternative, then it can be inferred that the alternative is much more likely to belong to this category. Assignment of such alternatives based on the probabilities without asking the DM refers to probabilistic classification (Buğdacı et al., 2013). Note that probabilistic classification of alternatives may lead to misclassification errors as in UTADIS. However, it decreases the cognitive burden of the DM by considering a small subset of the alternatives. In this section, we first present the proposed probabilistic algorithm and its steps. We then summarize the probabilistic approach of the benchmark study.

3.4.1 The proposed probabilistic algorithm

Once we define the possible category ranges of the alternatives through the mathematical models, we sum up the frequencies of hypothetical assignments of the alternatives and calculate the assignment probabilities. If the assignment probability of an alternative is higher than the critical value, $1-\tau$, for a category, then this alternative is probabilistically assigned to the corresponding category. However, when there is no assignment information obtained from the DM, the probabilistic assignments can result in high degree of misclassification. Different than the probabilistic algorithm of Buğdacı et al. (2013), $A_{PBU\breve{G}}$, we define a cut-off value to determine the allowance of the probabilistic assignments during the decision process. At each iteration, we calculate the average relative entropy (ARE) of C_0, the set of unlabeled alternatives as in (3.30).

$$ARE = \frac{\sum_{a_j \in C_0} H_R(a_j)}{|C_0|} \tag{3.30}$$

ARE stands for the average uncertainty of the unlabeled alternatives. If ARE is one, then each unlabeled alternative has equal probabilities over the possible categories. We define 0.5 as the critical value for allowing probabilistic assignments. In other words, if ARE is less than 0.5 at an iteration, then our algorithm makes probabilistic assignments of the alternatives with higher probability values than the critical value, $1-\tau$. Otherwise, we skip the probabilistic assignments at any iteration. We next provide the steps of the probabilistic algorithm.

3.4.1.1 Algorithm A_{PRENT}

Step 0: Set $C_0 = A$, $C_k = \emptyset$ for $k = 1, \dots, q$, $C_t^W = q$ and $C_t^B = 1$ for each $a_t \in C_0$.

Step 1: For each $a_t \in C_0$, set $k = C_t^B$.

1.1. Solve $(LP1_{a_t,k})$.

(i) If $obj_1^*(a_t, k) \geq 0$, set $C_t^W = k$. Set $C_{t'}^W = k$ for each alternative $a_{t'} \in C_0$ that dominates alternative a_t and if $C_{t'}^W = C_{t'}^B = k$, let $C_k \leftarrow C_k \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_t^W = C_t^B = k$, then let $C_k \leftarrow C_k \cup \{a_t\}$, $C_0 \leftarrow C_0 - \{a_t\}$ and go to Step 1.3. Otherwise set $k = k - 1$ and go to Step 1.2.

(ii) If $obj_1^*(a_t, k) < 0$ and $k < C_t^W - 1$, set $k = k + 1$ and repeat Step 1.1.

(iii) If $obj_1^*(a_t, k) < 0$ and $k = C_t^W - 1$, go to Step 1.2.

1.2. Solve $(LP2_{a_t,k})$.

(i) If $obj_2^*(a_t, k) < 0$, set $C_t^B = k + 1$. Set $C_{t'}^B = k + 1$ for each alternative $a_{t'} \in C_0$ that is dominated by alternative a_t and if $C_{t'}^W = C_{t'}^B = k + 1$, then let $C_{k+1} \leftarrow C_{k+1} \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_t^W = C_t^B = k + 1$, then let $C_{k+1} \leftarrow C_{k+1} \cup \{a_t\}$ and $C_0 \leftarrow C_0 - \{a_t\}$. Go to Step 1.3.

(ii) If $obj_2^*(a_t, k) \geq 0$ and $k > C_t^B$, set $k = k - 1$ and repeat Step 1.2.

(iii) If $obj_2^*(a_t, k) \geq 0$ and $k = C_t^B$, go to Step 1.3.

1.3. If the category range of all alternatives are reduced, then go to Step 2. Otherwise go to step 1.1 for the next $a_t \in C_0$.

Step 2: For each $a_t \in C_0$, if $ARE < 0.5$ and $p(x_{tk} = 1) \geq 1 - \tau$, then let $C_k \leftarrow C_k \cup \{a_t\}$ and $C_0 \leftarrow C_0 - \{a_t\}$. If $C_0 = \emptyset$, go to step 4. Otherwise go to Step 3.

Step 3: Select an alternative $a_t^s \in C_0$ to ask the DM. The DM assigns alternative a_t^s to $C_{k'}$, then let $C_{k'} \leftarrow C_{k'} \cup \{a_t^s\}$ and $C_0 \leftarrow C_0 - \{a_t^s\}$. Set $C_{t'}^W = k'$ for each alternative $a_{t'} \in C_0$ that dominates alternative a_t^s. Set $C_{t'}^B = k'$ for each alternative $a_{t'} \in C_0$ that is dominated by alternative a_t^s. If $C_{t'}^W = C_{t'}^B$, let $C_{k'} \leftarrow C_{k'} \cup \{a_{t'}\}$ and $C_0 \leftarrow C_0 - \{a_{t'}\}$. If $C_0 = \emptyset$, go to step 4. Otherwise go to Step 1.

Step 4: Present the categories of alternatives to the DM and stop.

Different than the non-probabilistic algorithm, we check whether the condition of probabilistic assignment is satisfied in Step 2 to assign the alternatives without asking the DM. If that is the case, then the alternatives are assigned to the categories whenever the probability of belonging to a category exceeds $1 - \tau$. In Step 3, we select the alternative with the highest relative entropy to be assigned by the DM. Figure 9 shows the flowchart of A_{PRENT}.

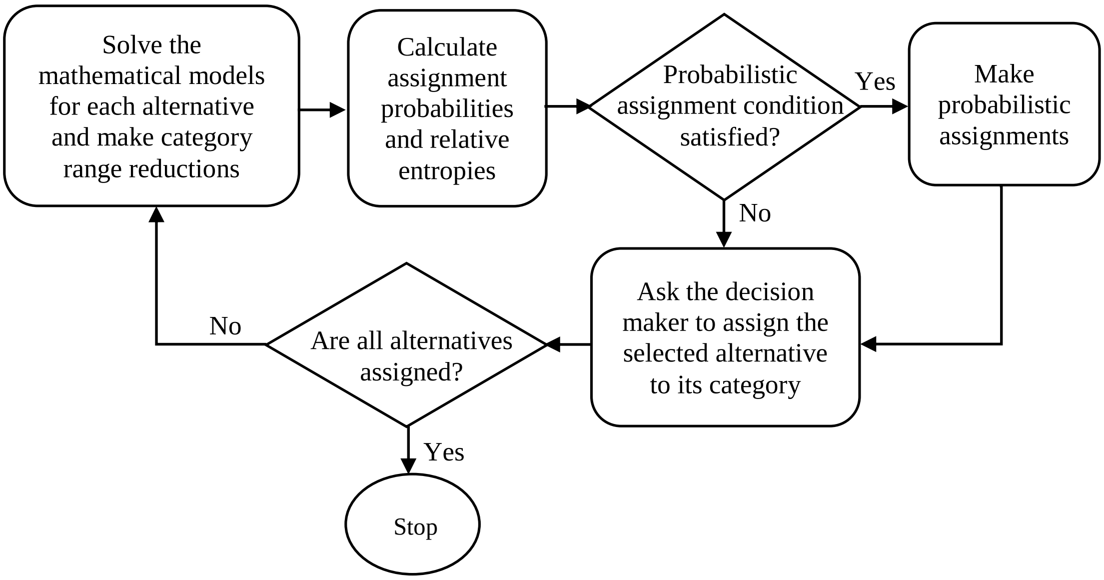


Figure 9. Flowchart of A_{PRENT}

3.4.2 Benchmark algorithm for probabilistic case

In order to test the performance of our probabilistic algorithms with relative entropy selection (A_{PRENTM} and A_{PRENTS}), the probabilistic algorithm of Buğdacı et al. (2013) ($A_{PBU\breve{G}}$) is used as benchmark. $A_{PBU\breve{G}}$ solves additional mathematical models to find the boundaries of the w_{ip} and u_k values. Assuming that their values show uniform distribution among their boundaries, the difference between the utilities of the alternatives and the category thresholds, $U(a_t) - u_k$, is assumed to follow a normal distribution based on the central limit theorem. Accordingly, they calculate $p(U(a_t) \geq u_k)$, the probability that the utility of alternative a_t is at least as much as the utility of the category threshold u_k.

Note that this probability measure is different from our probability definition where we calculate the probability of an alternative belonging to a category. To illustrate, let alternative a_t have a possible category range of $\{C_1, C_2, C_3\}$ in a four-category problem where C_1 is the most preferred category. Then, in the approach of Buğdacı et al. (2013), it must be the case that $p(U(a_t) \geq u_3) = 1$ since C_4 is not in the possible category range. Moreover, if $p(U(a_t) \geq u_1) = 0.7$, then it must be the case that $p(U(a_t) \geq u_2) \geq 0.7$ since $u_2 < u_1$. In our approach, on the other hand, the assignment probability of alternative a_t to C_4, $p(a_t \in C_4)$, is equal to zero and the summation of the probabilities for the rest of the categories is equal to one.

$A_{PBU\breve{G}}$ selects the alternative to be asked the DM that has the closest probability value to 0.5 for a category as in (3.25). Such a selection process considers a single category and ignores the distribution of probabilities among each category. For instance, let alternative a_1 and a_2 have the category ranges of $\{C_1, C_2, C_3\}$ and $\{C_1, C_2\}$, respectively. Let $p(U(a_1) \geq u_1) = 0.33$, $p(U(a_1) \geq u_2) = 0.67$ and $p(U(a_2) \geq u_1) = 0.40$. Then the algorithm selects alternative a_2 since its probability value is closer to 0.5 than that of alternative a_1. However, alternative a_1 has similar assignment probability values for the three categories and hence its assignment uncertainty may be higher than that of alternative a_2.

According to the user-defined critical probability threshold, τ, probabilistic assignments are made as follows:

- If $p(U(a_t) \geq u_1) \geq 1 - \tau$, then a_t is assigned to C_1;
- If $p(U(a_t) \geq u_k) \geq 1 - \tau$ and $p(U(a_t) \geq u_{k-1}) \leq \tau$, then a_t is assigned to C_k;
- If $p\big(U(a_t) \geq u_{q-1}\big) \leq \tau$, then a_t is assigned to C_q.

At each iteration, the LP models are solved to identify the category ranges of the alternatives. Then, additional LP models are solved to define the probability that the utility of an alternative is larger than a category threshold. The alternatives are probabilistically assigned to categories and then the algorithm selects the alternative that has a probability value closest to 0.5 to ask the DM. The assignment information obtained from the DM is added to the LP models in the upcoming iterations.

CHAPTER 4

COMPUTATIONAL EXPERIMENTS

We implement the proposed algorithms and benchmark algorithms on three example problems from the literature in Sections 4.1, 4.2 and 4.3, and on randomly generated problems in Section 4.4. For the first two problems, we make experiments on non-probabilistic case for both unconstrained and constrained settings where in the latter category size restrictions and initial assignment examples are considered. In the third problem and the randomly generated problems, we consider the non-probabilistic and probabilistic algorithms together. We assume piecewise linear preference functions in the first and third problems whereas the second problem considers general monotone preference functions. Finally, we incorporate each additive function in the randomly generated problems. We use 4 GHz, 8 GB of RAM, Intel Core i7 computer to perform the all experiments. The mathematical models are solved using CPLEX 12.5 solver.

4.1 Applications on MBA problem

Financial Times (FT) regularly announces the rankings of the MBA, executive MBA and master programs based on several criteria such as salary % increase and faculty with doctorates. Some criteria (e.g., alumni recommend) are announced on an ordinal scale while the others have ratio scales. Köksalan et al. (2010) study the 2005 rankings of global MBA programs to implement their approach using weight ranges instead of fixed weights. They convert the ordinal-based criteria into standardized ratio scale using an LP. The same data is used by Köksalan and Özpeynirci (2009) and Buğdacı et al. (2013). We apply our algorithm to the same sorting problem of 81 MBA programs evaluated on three main criteria: alumni career progress, diversity, and idea generation. The 81 MBA programs are placed into three categories assuming that the

DM has an underlying additive preference function in piecewise linear form. The w_{ip} and u_k values to find the true categories of the alternatives are obtained from the three studies mentioned above. The marginal preference functions in each criterion is presented in Figure 10. The number of subintervals is three for each criterion. For example, the DM assigns 0.24 utility for the score of 0.2 in criterion 2 while the utility of the score of 0.8 in criterion 3 is approximately 0.15.

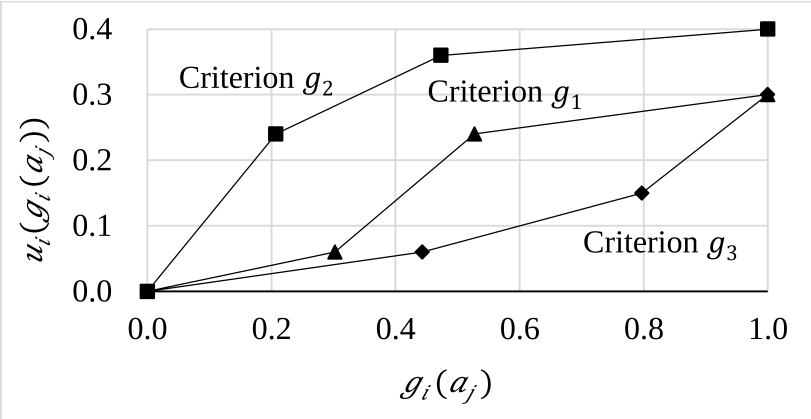


Figure 10. The marginal preference functions of criteria in MBA problem

Table 1 shows the w_{ip} values for each subinterval and criterion. The w_{ip} values are identified to reflect different types of preference structures (Köksalan and Özpeynirci, 2009). The utility thresholds for categories have the following values: $u_1 = 0.65$ and $u_2 = 0.4$. According to the underlying preference structure, 15, 47, and 19 alternatives belong to categories 1, 2, and 3, respectively. Note that we assume that the w_{ip} and u_k values are unknown and used to identify the categories of the alternatives that are elicited from the DM.

Table 1. w_{ip} values that represent the DM's preferences in MBA problem

	p		
i	1	2	3
1	0.06	0.18	0.06
2	0.24	0.12	0.04
3	0.06	0.09	0.15

4.1.1 Unconstrained case in MBA problem

We first apply the algorithms on MBA programs data without any category size restriction or initial assignments of the DM. We first provide the assignments in iterations of A_{RENTM} and then summarize the results for all algorithms. Table 2 shows the assignments and category range reductions made by the models as well as the DM's assignments including the alternative number, its category range and the exact category. In the first iteration, there are 81 alternatives that are not placed to any category and two thresholds separating the categories. After solving the two models, we make a hypothetical assignment for each alternative. Since there are two category thresholds, the algorithm checks the assignment of each alternative to categories $162\ (81 \times 2)$ times in total.

Since there is no initial assignment, the models cannot narrow down the category range of any alternative in the first few iterations and only dominance relations contribute to the category range reductions. In the first iteration, alternative 68 has the maximum relative entropy score among all alternatives and it is chosen for the DM's assignment. The DM places it into the 3rd category. The selected alternative does not lead to any category range reduction either by models or the dominance relations.

In the second iteration, alternative 11 is selected and the DM places it into the 1st category. Since alternative 5 dominates alternative 11, alternative 5 is also assigned to the 1st category by the dominance check. After the third iteration, the algorithm starts to select the alternative from narrower category range to ask the DM and this continues until the algorithm terminates. Iteration 12 is the first iteration where the models assign an alternative even though there is no dominance relation with the alternatives that are previously assigned by the DM. The algorithm continues until all alternatives are assigned to their exact categories in iteration 38.

Table 2. Assignments of A_{RENTM} on MBA problem

Iter. No.	Assignments by the models			Category range reductions		DM's assignments		
	C_1	C_2	C_3	$\{C_1, C_2\}$	$\{C_2, C_3\}$	Alt.	Cat. range	Cat.
1						68	1-3	3
2	5					11	1-3	1
3				7, 18, 19, 21, 24, 28, 33, 34, 36, 45, 52	57, 60, 63, 65, 66, 69, 70, 71, 72, 73, 75, 76, 79, 80, 81	44	1-3	2
4		52			53, 59, 61, 62, 67, 74	60	2-3	2
5						24	1-2	2
6						59	2-3	3
7						63	2-3	3
8				1, 3, 13, 20, 37, 38, 42	64, 77	61	2-3	2
9					46, 51, 55, 56	73	2-3	3
10			80			3	1-2	1
11				2, 4, 9, 14, 16, 23, 35, 39, 47		69	2-3	2
12		37, 57	66, 72	32		51	2-3	3
13			77		41, 48	1	1-2	1
14						54	2-3	2
15						81	2-3	3
16			64		78	18	1-2	1
17		53, 62		8		74	2-3	2
18				30	43	30	1-2	2
19		32, 38, 43			31, 49	58	2-3	2
20	2, 4				50	14	1-2	1
21						41	2-3	3
22			55		17, 27	8	1-2	2
23	7, 13	34, 49		40	26, 29	35	1-2	2
24		26, 29, 42, 47			15	17	2-3	2
25	19	15, 31, 40		6, 10, 12, 22, 25		21	1-2	1
26		45, 65				79	2-3	2
27		33, 70				78	2-3	2
28		36, 76	48, 56, 71			12	1-2	1
29	9	25				16	1-2	2
30		22				46	2-3	3
31			50			75	2-3	3
32		23				10	1-2	2
33						28	1-2	2
34						6	1-2	1
35						20	1-2	2
36						27	2-3	2
37						39	1-2	2
38						67	2-3	2

We also apply the benchmark algorithms and selection methods on MBA problem. We expect that the DM spends less effort while assigning the alternatives with narrower category ranges compared to the ones that have wider category ranges. In order to compare the algorithms in terms of the assessment burden on the DM, we introduce a new measure, the number of pairwise category decisions (PCD). Each category range reduction by the DM is regarded as a PCD. That is, an assignment of the DM among k categories is regarded as $k-1$ PCDs. Let AS_k be the number of assignments of the DM among k possible categories. Total number of PCDs in a q-category problem is defined as follows:

$$PCD = \sum_{k=2}^{q} (k-1) * AS_k \tag{4.1}$$

If, for example, the number of DM's assignments among two and three categories are α and β, respectively, then the number of pairwise category information required from the DM is calculated as $\alpha + 2\beta$. We note that for an alternative the probabilities of belonging to categories may differ. However, in this study, regardless of these probabilities we focus on the number of possible categories that the DM needs to select in order to place the alternative.

The numbers of assignments by each algorithm are reported in Table 3. $A_{\text{Ö}ZP}$ is the worst performing algorithm among six algorithms. The alternatives, i.e., the MBA programs, in the first and third categories have higher number of dominance relations than the ones in the second category. This causes $A_{\text{Ö}ZP}$ to ask the DM alternatives from the first and third categories in the early iterations. However, approximately 60% of the alternatives belong to the second category in MBA programs dataset. $A_{\text{Ö}ZP}$ could not narrow down the category ranges of the alternatives in the second category until 24 alternatives are assigned by the DM. That is why the selection method of higher cardinality causes to frequently ask the DM among wider category ranges and this results in poor performance compared to the other algorithms including the random selection.

Algorithms A_{REG}, $A_{BU\text{Ğ}}$, A_{RENTM} and A_{RENTS} perform better than A_{RAND} in terms of the assignments made by models. $A_{BU\text{Ğ}}$ assigns one more alternative than A_{REG} while requiring higher computational effort than A_{REG}. Since no preference information is imposed to the LPs in the first iteration of each algorithm, the objective function values are identical for each alternative. That is why the regret scores of all alternatives in the first iteration are same in A_{REG}, i.e., the selection method could not identify an alternative and hence, an alternative is randomly selected to be assigned by the DM. As in A_{RAND}, we generate 100 random samples for the selection process to break the ties in the regret scores of A_{REG}. The average of the assignments based on 100 random samples is reported in A_{RAND} and A_{REG}.

Table 3. Performances of the algorithms in MBA problem - unconstrained case

Algorithm	Assignments by the models	DM's assignments		Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories			
A_{RAND}	25.5	45.8	9.7	65.2	5912	89
$A_{ÖZP}$	16	27	38	103	9491	172
A_{REG}	40	33.9	7.1	48.1	4882	86
$A_{BUĞ}$ [1]	41	34	6	46	8261	140
A_{RENTM}	43	35	3	41	3893	79
A_{RENTS}	46	32	3	38	1147	221

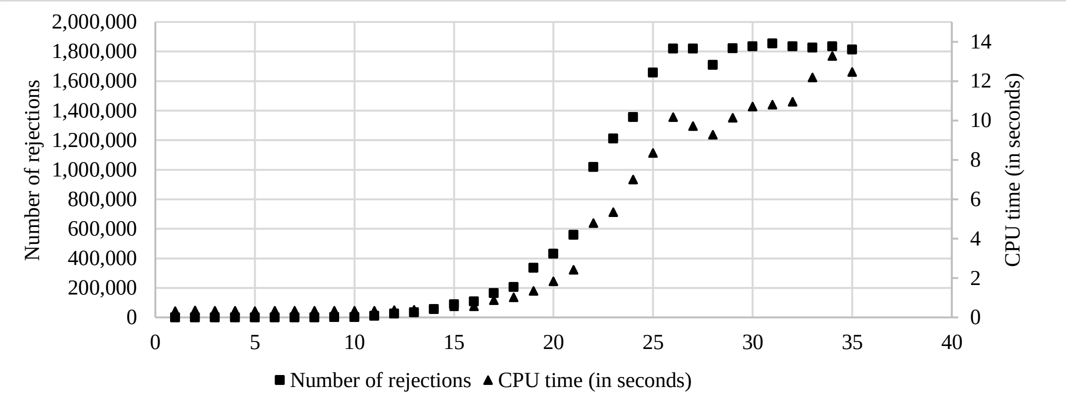
The best algorithms in terms of the number of questions asked are our algorithms, A_{RENTM} and A_{RENTS}. The two algorithms tend to select the alternatives to ask the DM from the narrower category ranges. Thus, they require less pairwise category information from the DM. Moreover, these two algorithms solve the least number of models to complete the classification among all algorithms. A_{RENTM} requires the DM to assign three more alternatives than A_{RENTS} while solving higher number of LP models than A_{RENTS}. However, parameter generation process of A_{RENTS} causes the algorithm to have longer CPU time than the other algorithms.

Figure 11 shows the number of rejections among the generated sets of parameters and their CPU time in each iteration. At the beginning of the fifth iteration, each category

[1] Due to round-off differences in the dataset, the results obtained by the algorithm of Buğdacı et al. (2013) are not identical to the results presented in Table 2 of Buğdacı et al. (2013).

has at least one alternative assigned either by the DM or the models, hence the algorithm starts to reject the randomly generated set of parameters due to incompatibility with the assignments of the DM. The parameter generation process starts to take longer than one second when the number of rejections increases to approximately 200,000. After the DM assigns 25 alternatives to categories, the number of rejections follow a steady path throughout ten more iterations. In general, the parameter generation process takes 148.88 seconds in total while the rest of the algorithm takes 72.28 seconds which is slightly less than the CPU time of A_{RENTM}.

Figure 11. Rejection rates of A_{RENTS} in MBA problem – unconstrained case



4.1.2 Constrained case in MBA problem

In this section, we consider the category size restrictions and initial assignments in MBA problem. As category size constraints, the sizes of the categories are given by the DM. Moreover, the DM initially assigns alternatives 14 and 21 to the first category, alternatives 10 and 79 to the second category, and alternatives 50 and 56 to the third category. In order to shorten the process, we utilize the dominance relations for narrowing down the possible assignments before solving any MIP model in each algorithm.

The performances of the algorithms are reported in Table 4. The initial assignments of the DM are reported in the assignments among three categories. It can be observed from Table 4 that there is a significant improvement in each algorithm with the inclusion of category size restrictions. A_{RAND} and $A_{ÖZP}$ have similar performances in terms of the number of assignments and PCDs. In $A_{ÖZP}$, asking the alternatives with higher dominance relations contribute to the category range reductions of the unlabeled alternatives. Hence, the number of MIPs solved in $A_{ÖZP}$ is less than that of A_{RAND}.

Table 4. Performances of the algorithms in MBA problem - constrained case

Algorithm	Assignments by the models	DM's assignments		Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories			
A_{RAND}	43.6	30.4	7	44.4	1992	442
$A_{ÖZP}$	45	28	8	44	1636	348
A_{REG}	56	18	7	32	1136	307
$A_{BUĞ}$	54	20	7	34	2717	555
A_{RENTM}	60	15	6	27	1048	210
A_{RENTS}	52	23	6	35	548	226

A_{REG} assigns two more alternatives than $A_{BUĞ}$ while solving less models. Since there are initial assignments of the DM, the regret scores of the alternatives differ starting from the first iteration. Therefore, it is not required to generate samples for the selection process in A_{REG}. A_{RENTS} requires the DM to assign higher number of alternatives than A_{REG} and $A_{BUĞ}$ while solving less number of models. A_{RENTM}, on the other hand, performs better than the benchmark algorithms and A_{RENTS} in terms of the amount of information obtained from the DM. Although A_{RENTS} takes a considerable amount of time in generating random set of parameters, the algorithm solves less number of models and hence, the CPU time of the two algorithms A_{RENTM} and A_{RENTS} are close to each other. We note that the computational time is higher for each algorithm when compared to the unconstrained version as MIPs are solved instead of LPs.

4.2 Applications on bus revision problem

In this section, we test the non-probabilistic algorithms on the bus revision problem that is used by Greco et al. (2010) and Özpeynirci et al. (2018). A transport company is responsible for the classification of 76 buses into four categories in order to identify whether the buses need major, minor or no revision. The buses that need major revisions are either in the worst (C_4) or lower-intermediate (C_3) technical state whereas the buses that require minor revisions are in the upper-intermediate (C_2) technical state. The most preferred category, (C_1) is the good technical state where the buses do not need any revision. The buses are evaluated on eight criteria defined in Table 5. There are four cost-related criteria where the lower scores are preferred to higher scores, and four gain-related criteria where higher scores are favored.

Table 5. Descriptions of criteria of bus revision problem

Criterion	Name	Type
g_1	Maximum speed	Gain
g_2	Compression pressure	Gain
g_3	Blacking	Cost
g_4	Torque	Gain
g_5	Summer fuel consumption	Cost
g_6	Winter fuel consumption	Cost
g_7	Oil consumption	Cost
g_8	Horse power	Gain

We assume that the preferences of the DM are consistent with a monotone non-decreasing additive utility function rather than a piecewise linear function. We solve $LP3_{a_t,k}$ and $LP4_{a_t,k}$ for defining the possible assignments of the alternatives when no category size restriction is imposed to the problem setting. In constrained problem, we solve $MIP3_{a_t,k}$ and $MIP4_{a_t,k}$ to identify the category ranges of the alternatives.

4.2.1 Unconstrained case in bus revision problem

Table 6 summarizes the performances of each algorithm for the bus revision problem without any category size restriction or initial assignments of the DM. Note that A_{RENTS} is not applicable in bus revision problem with general monotone case, thus, we report the results of five algorithms. $A_{\text{ÖZP}}$ and $A_{BU\breve{G}}$ consult the DM more to assign the alternatives when compared to other algorithms. Moreover, the number of models solved in these two algorithms is much higher than that of other algorithms. A_{RAND} and A_{REG} perform similarly in terms of the number of questions asked and the number of models solved. However, A_{RAND} selects the alternatives from narrower category ranges and hence, results in less number of PCDs. A_{RENTM} tends to select alternatives from narrower category ranges and asks the least number of questions to the DM with a slight increase in the number of models solved compared to those of A_{RAND} and A_{REG}. The CPU times of the algorithms are in parallel with the number of models solved.

Table 6. Performances of the algorithms in bus revision problem-unconstrained case

Algorithm	Assignments by the models	DM's assignments			Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories			
A_{RAND}	27.3	24.2	14.9	9.6	82.8	6574	135
$A_{ÖZP}$	17	9	21	29	138	11067	210
A_{REG}	27.1	22.1	15.6	11.2	86.9	6533	133
$A_{BUĞ}$	21	15	16	24	119	12046	241
A_{RENTM}	29	29	12	6	71	6758	135

4.2.2 Constrained case in bus revision problem

We assume the sizes of the categories 1 to 4 are given by the DM as 14, 18, 22 and 22, respectively and six alternatives are initially assigned to categories by the DM as in Özpeynirci et al. (2018). Table 7 shows the iterations of A_{RENT} on bus revision data with category size restrictions and initial assignments. The assignments to four categories at iteration zero are the initial assignments of the DM. Three alternatives are assigned to the 4^{th} category and one alternative is assigned to each of the other categories. Since six alternatives are initially assigned by the DM, 70 alternatives are left to be assigned to their exact categories. In the first iteration, all alternatives in the 1^{st} category are identified through the MIP models. Hence, the 1^{st} category is eliminated from the range of possible categories for the rest of the alternatives. 31 alternatives are assigned by the models in the first iteration. Alternative 21 has the maximum relative entropy score and is selected to be assigned by the DM.

Table 7. Iterations of A_{RENT} on bus revision data

Iter. No.	Assignments by the models				Category range reductions			DM's assignments		
	C_1	C_2	C_3	C_4	$\{C_2, C_3\}$	$\{C_2, C_3, C_4\}$	$\{C_3, C_4\}$	Alt.	Cat. range	Cat.
0	33	64	13	2, 67, 70						
1	7, 18, 29, 37, 44, 49, 55, 57, 61, 65, 71, 72, 76	4, 5, 32, 35, 41, 42, 51, 54, 56, 59, 73, 74, 75	1, 15, 17, 20, 22	6, 10, 14, 19, 23, 27, 30, 34, 39, 40, 45, 47, 48, 50, 60, 62, 63, 69,	16, 21, 25, 31, 43, 52, 58, 73	9, 26, 38, 68	3, 8, 11, 12, 24, 28, 36, 46, 53, 66	21	2-3	3
2			25, 53, 58				26, 38	8	3-4	3
3								31	2-3	3
4			11, 26, 38					52	2-3	2
5								9	2-4	2
6								16	2-3	2
7			12, 24, 28, 36, 46, 66					3	3-4	4
8		68						43	2-3	3

After the DM assigns alternative 21 to the 3^{rd} category, the MIP models assign three alternatives to Category 3 and the possible categories of two alternatives are reduced. Then the DM assigns alternative 8 and 31 to the 3^{rd} category and the MIPs assign three more alternatives to the 3^{rd} category. Afterwards, three alternatives are assigned to the 2^{nd} category by the DM. In iteration 7, the DM assigns alternative 3 to the 4^{th} category which reaches the category size limit with 22 alternatives. The elimination of the 4^{th} category leads to the assignment of six alternatives to the 3^{rd} category. After the DM assigns alternative 43 to the 3^{rd} category, the remaining alternative 68 is automatically assigned to the 2^{nd} category and the algorithm terminates.

The assignment performances of each algorithm are reported in Table 8. Since the 1^{st} category is eliminated from the range of possible categories at the beginning of the first iteration, the DM's assignments among four categories in the fifth column include the initial assignments. $A_{BUĞ}$ performs worse than the other algorithms in terms of the information obtained from the DM and the computational effort. A_{RAND} and $A_{ÖZP}$ have similar performances in terms of the number of assignments. A_{RAND} selects the alternatives from the narrower range, hence it outperforms $A_{ÖZP}$ in terms of the number of PCDs.

As in the MBA problem, no random sampling is required in A_{REG} since initial assignments lead to different regret scores for each alternative in each iteration. A_{RENTM} and A_{REG} ask the same number of questions with the same category ranges. The reason for A_{REG} to solve less number of models is that A_{REG} eliminates the 4^{th} category from the possible categories of the unlabeled alternatives in the second iteration whereas A_{RENTM} does this in iteration 7. This is also reflected in the CPU times of the two algorithms to complete the task.

Table 8. Performances of the algorithms in bus revision problem - constrained case

Algorithm	Assignments by the models	DM's assignments			Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories			
A_{RAND}	58.9	9.8	1.3	6	30.4	310	71
$A_{ÖZP}$	59	7	4	6	33	353	74
A_{REG}	62	7	1	6	27	206	45
$A_{BUĞ}$	55	12	3	6	36	1328	254
A_{RENTM}	62	7	1	6	27	296	64

4.3 Applications on Energy Trilemma Index problem

We apply the non-probabilistic and probabilistic algorithms on energy data without any category size restrictions or initial assignments. We assume that the preferences of the DM are consistent with a piecewise linear preference function. In cooperation with the global consultancy company, Oliver Wyman, the World Energy Council (WEC) yearly announces the Energy Trilemma Index (ETI) to demonstrate the energy performance of countries (WEC, 2019). In the index data announced in 2019, four main criteria are used to measure the energy performance of 128 countries where three criteria (energy security, energy equality and environmental sustainability) are directly related to energy and one criterion is related to the general condition of the countries.

Many sub-criteria play role to generate the four main criteria values. Import dependency, electricity generation diversity and energy storage capacity measures are used to measure energy security. The second criterion, energy equity, is measured using access to electricity and clean cooking, electricity prices and gasoline and diesel prices. In environmental sustainability, factors such as energy density, low carbon electricity production and per capita carbon dioxide (CO2) emission criteria are used as sub-criteria. Macroeconomic stability, effectiveness of the government and innovation capability are some sub-criteria to represent the general country situation. In Figure 12, three energy-related criteria are shown in form of triple trivet.

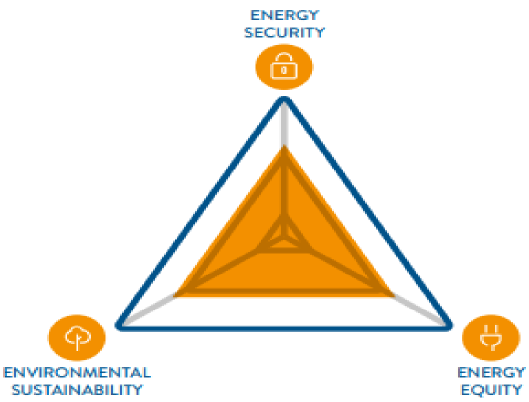


Figure 12. The triple trivet of the ETI (WEC, 2019)

WEC calculates the aggregate score of the countries by using fixed weights, 30% for each of the three energy-related criterion and 10% for the general condition criterion. 128 countries are ranked based on their aggregate scores. WEC also assigns countries into four categories each of which includes 32 countries. The first category involves the countries with the highest energy scores while the fourth category includes the countries with the lowest energy performance. The classification of the countries based on their enegy performances plays an important role in the assesment of the sustainability of national energy policies.

Table 9. w_{ip} values that represent the DM's preferences in ETI problem

	p		
i	1	2	3
1	0.17	0.09	0.04
2	0.07	0.14	0.09
3	0.10	0.07	0.13
4	0.05	0.03	0.02

When looking at the minimum and maximum scores in the criteria, the range in the energy security is [30,79] while the scores in the energy equity are in the range of [5,100]. In all criteria, the scores are scaled to the range of [0,1] with the lowest score being zero and the highest score being one. We use the scaled data of ETI problem to apply the proposed and benchmark algorithms. We use three subintervals to represent the preferences of the DM in each criterion. Table 9 shows the w_{ip} values for each subinterval and criterion in ETI problem. The marginal preference functions in each criterion is presented in Figure 13. The sum of w_{ip} values in each energy-related criterion is set as 30% and the total weight of the fourth criterion is set to 10%. w_{ip} values are determined in a way that the weights of different subintervals in each energy-related criterion are higher than those of other subintervals. For example, the weight of the first subinterval in the first criterion is higher than that of other subintervals while the weights of the second and third subintervals are higher in the second and third criteria, respectively.

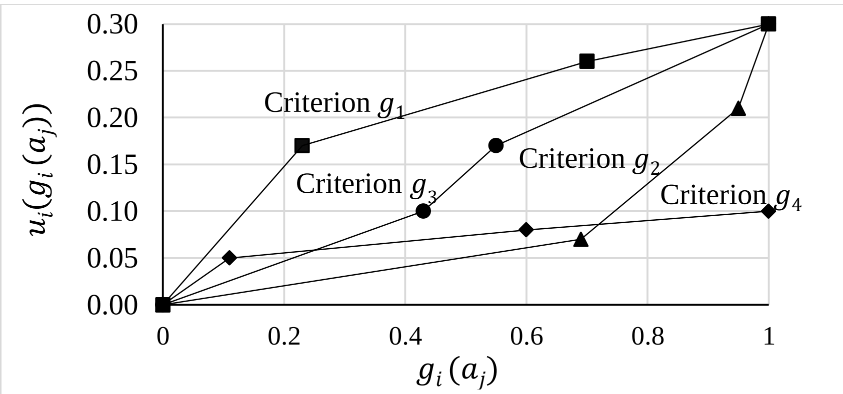


Figure 13. The marginal preference functions of criteria in ETI problem

In order to find the categories of the alternatives, the category thresholds are determined as $u_1 = 0.73$, $u_2 = 0.60$ and $u_3 = 0.47$. According to the category thresholds and the aggregate utility of the alternatives calculated with respect to the preference structure of the DM presented in Figure 13. 28, 33, 36 and 31 alternatives are assigned to the first, second, third and fourth categories, respectively. In Table 10, the countries in each category are presented in a decreasing order. For example, Switzerland is the country with the best energy performance in the first category, while Australia is the worst in energy performance in the same category.

Table 10. Country classifications based on energy performance

Cat.	Countries
C_1	Switzerland, Luxembourg, Denmark, Sweden, UK, Austria, France, Norway, Netherlands, Finland, Iceland, Slovenia, New Zealand, Germany, United States, Ireland, Italy, Spain, Canada, Uruguay, Czech Republic, Hungary, Belgium, Israel, Slovakia, Croatia, Hong Kong, Australia
C_2	Latvia, Argentina, Japan, Portugal, Romania, Malta, Estonia, Singapore, Korea, Lithuania, Costa Rica, Bulgaria, Brazil, Mexico, Greece, Venezuela, United Arab Emirates, Russia, Qatar, Panama, Ecuador, Chile, Cyprus, Mauritius, Colombia, Brunei, Oman, Poland, Kuwait, Azerbaijan, Malaysia, Peru, Armenia
C_3	El Salvador, Bahrain, Montenegro, Paraguay, Turkey, Kazakhstan, Saudi Arabia, Ukraine, Namibia, Thailand, Northern Macedonia, Sri Lanka, Trinidad and Tobago, Georgia, Iran, Indonesia, Serbia, Tunisia, the Philippines, China, Guatemala, Albania, Morocco, Bolivia, Bosnia and Herzegovina, Angola, Lebanon, Algeria, Kenya, Myanmar, Egypt, Botswana, Gabon, Ghana, Vietnam, Iraq
C_4	Zambia, Tajikistan, Honduras, Nicaragua, South Africa, Jordan, Eswatini, Ivory Coast, Cambodia, Madagascar, Malawi, Cameroon, Pakistan, Mozambique, India, Zimbabwe, Mauritania, Tanzania, Moldova, Bangladesh, Ethiopia, Mongolia, Jamaica, Dominican Republic, Senegal, Nigeria, Benin, Chad, Congo, Nepal, Niger

4.3.1 Non-probabilistic case in ETI problem

Table 11 shows the assignment performances of the algorithms in the ETI problem for the non-probabilistic case. $A_{ÖZP}$ and A_{RAND} have similar assignment performances while $A_{ÖZP}$ selects the alternatives to ask the DM among wider category ranges. $A_{BUĞ}$, on the other hand, performs better than $A_{ÖZP}$ and A_{RAND}. However, the number of models solved in $A_{BUĞ}$ is about 50% more than the two algorithms. A_{REG}, A_{RENTM} and A_{RENTS} elicit less amount of assignment information from the DM when compared to the other algorithms. A_{REG} asks the DM two more assignments than

A_{RENTM} and A_{RENTS} while the number of models solved by A_{REG} and A_{RENTM} is close to each other. Our algorithms, A_{RENTM} and A_{RENTS} require the DM to assign the same number of alternatives. A_{RENTS} selects the alternatives to ask the DM among narrower category ranges than A_{RENTM} and this is reflected in the number of PCDs. Furthermore, A_{RENTS} solve less number of models to define the category range of the alternatives since the algorithm first makes hypothetical assignments and then solves models if necessary, i.e., if no hypothetical assignment is made in a category. However, A_{RENTS} takes excessive amount of time to complete the task due to random data generation process.

Table 11. Performances of the algorithms for non-probabilistic case in ETI problem

Algorithm	Assignments by the models	DM's assignments			Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories			
A_{RAND}	48.4	56	15.3	8.3	111.5	16474	318
$A_{ÖZP}$	47	45	11	23	125	15651	285
$A_{BUĞ}$	59	50	12	7	95	25231	514
A_{REG}	65	41	16	6	91	15642	294
A_{RENTM}	67	48	9	4	78	15369	289
A_{RENTS}	67	54	5	2	70	3676	6520

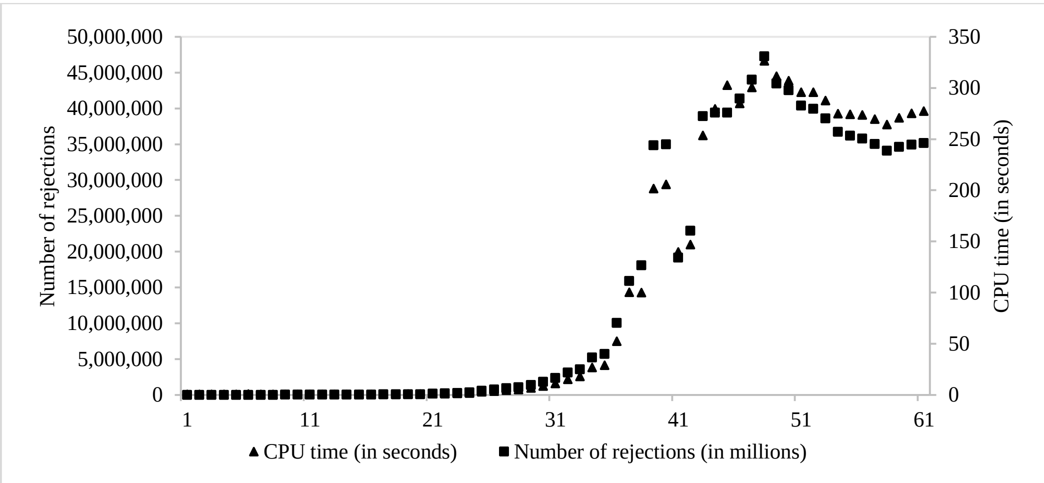


Figure 14. Rejection rates of A_{RENTS} for non-probabilistic case in ETI problem

Figure 14 shows the iteration based number of rejections and their corresponding CPU times in randomly generated sample set of parameters when A_{RENTS} is applied in ETI problem. The number of rejections reaches to a million in iteration 28 of A_{RENTS} while it takes 5 seconds to compete the data generation process. The rejection rates continue to increase until iteration 48 where the sample generation process takes 326.5 seconds. Afterwards, the number of rejections starts to decrease while the CPU times stay at a certain level.

In Figure 15, ARE values are given throughout iterations of A_{RENTM} in the ETI problem for the non-probabilistic case. Having a score of 0.73 in the first iteration, ARE suddenly falls to 0.53 with three assignment information received from the DM. After making a sharp rise to 0.63, it follows a downward trend with small increases in some iterations. In the 15^{th} iteration, it is seen that ARE value is below 0.50. After moving in the 0.45–0.50 range, it continues the downward trend and reaches the zero level in the 49^{th} iteration. In the rest of the iterations, each unlabeled alternative is hypothetically assigned to a single category and hence ARE values remain at zero.

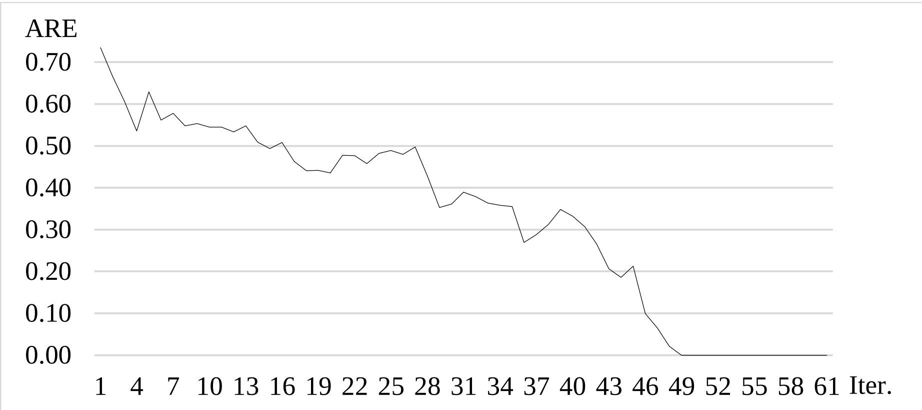


Figure 15. ARE values throughout the iterations in A_{RENTM} in ETI problem

4.3.2 Probabilistic case in ETI problem

In the probabilistic case, the alternatives are assigned to categories when the assignment probability exceeds a predetermined threshold value, τ. The proposed probabilistic algorithms, A_{PRENTM} and A_{PRENTS} allow the probabilistic assignments when ARE value is less than 0.5. A_{PRENTM}, A_{PRENTS} and $A_{PBUĞ}$ are applied on ETI problem for different τ values in the next section. Then we consider the case of no τ value and represent the results of A_{PRENTM} and A_{PRENTS} in Section 4.3.2.2.

4.3.2.1 Probabilistic assignments with different τ values in ETI problem

Table 12 shows the probabilistic and non-probabilistic assignments in iterations of A_{PRENTM} in case of $\tau = 0.05$ in ETI problem. The probabilistic assignments are presented in italic font while misclassified alternatives are shown in bold font. ARE values and the assignments of the DM's are given for each iteration. For the sake of brevity, the category range reductions are not presented in Table 12.

In the first eight iterations, the possible category ranges are narrowed down by the LP models. Three alternatives are assigned to their exact categories by the models in the 9th iteration. It is seen that ARE falls below 0.5 for the first time in 15th iteration. This shows that the necessary condition for making probabilistic assignments is met. Since τ = 0.05, alternatives having assignment probabilities of at least 0.95 are assigned probabilistic. ARE of the 19 alternatives assigned probabilistically is around 0.12 in the 15th iteration. Hence, the removal of these alternatives from the system causes a rise in the ARE value in the upcoming iteration. The probabilistic assignments are performed in the 28th, 37th, 40th and 43rd iterations. In the 40th iteration, alternative 49 was probabilistically assigned to the second category while this alternative belongs to the first category. The second misclassification is made in the last iteration with alternative 45.

Assignment performances of the probabilistic algorithms, $A_{PBUĞ}$, A_{PRENTM} and A_{PRENTS} are presented for different τ values in the ETI problem in Tables 13, 14 and 15, respectively. In order to compare the probabilistic case with the non-probabilistic case, the assignment performances of the non-probabilistic algorithms are given as well. The number of assignments (i) by the models, (ii) made probabilistically and (iii) by the DM are reported in separate columns. The number of misclassification errors and their distributions among the categories are given in a way that if an alternative in the first category is assigned to the second category, then this is recorded in the first category as a misclassification error.

Table 12. Assignments of A_{PRENTM} in ETI problem, $\tau = 0.05$

Iter.	Assignments made probabilistically or by the models					DM's assignments		
No.	C_1	C_2	C_3	C_4	ARE	Alt.	Range	Cat.
1					0.73	3	1-4	3
2					0.67	32	1-3	1
3					0.61	22	1-4	4
4					0.54	121	1-4	2
5					0.63	67	1-4	3
6					0.56	84	3-4	4
7					0.58	101	1-3	2
8					0.55	69	1-2	1
9	113			12, 88	0.55	52	2-4	4
10					0.54	33	2-4	4
11					0.54	1	2-4	3
12					0.53	94	2-3	2
13					0.55	51	1-2	1
14	112				0.51	124	1-2	1
15	7, *40, 41, 44, 58, 85, 86, 108, 122*	*16*		*10, 26, 39, 77, 78, 89, 93, 104, 115, 128*	0.49	107	1-2	1
16					0.58	24	2-4	3
17					0.52	9	2-4	3
18					0.52	92	2-3	2
19					0.51	65	2-3	2
20			35		0.51	55	3-4	3
21					0.51	5	2-3	2
22					0.52	28	1-2	1
23	50				0.52	66	1-2	2
24					0.51	37	1-2	2
25	123	18, 25, 76			0.56	57	1-2	1
26	56		118		0.55	62	2-3	3
27			54		0.55	19	3-4	4
28	*30, 91, 110*	*8, 17, 23, 29, 34, 46, 64, 75, 98, 100, 125*	*2, 13, 105, 117*	20, *59, 74*	0.48	4	1-2	2
29					0.62	116	2-3	3
30					0.6	61	3-4	4
31					0.58	6	1-2	1
32					0.53	42	3-4	3
33					0.52	102	2-3	2
34					0.52	119	2-3	3
35			90		0.55	96	2-3	2
36			14		0.53	60	1-2	2
37	21	27, 68, *73, 106*	*43, 47, 53*, 80, *97, 103, 111*	*38, 81*	0.47	79	2-3	3
38					0.62	63	3-4	3
39					0.5	114	3-4	4
40	*11*	**49,** *72*	*82, 95, 120, 126*	*48, 71, 87*	0.37	99	1-2	2
41					0.74	36	2-3	3
42					0.63	127	3-4	4
43			*83*	*31,* **45,** *70, 109*	0.31	15	3-4	3

A new measure, the number of category misclassifications (CMs), is introduced to check how much a probabilistic assignment is misclassified. Each misclassification of one category, regardless of being assigned to lower/upper category, is counted as one CM. For example, if an alterantive belongs to the first category and probabilistically assigned to the third category, then this is regarded as 2 CMs. Finally, the correctly assigned alternatives among the probabilistically assigned ones are presented.

Table 13. Performance of $A_{PBU\breve{G}}$ in ETI problem

τ	Assignments			# of errors	# of CMs	Dist. of errors				# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM			C_1	C_2	C_3	C_4				
Non-prob.	59	-	69	-	-	-	-	-	-	-	-	25231	514
0.05	29	36	63	0	0	0	0	0	0	36	100.0	21863	465
0.10	14	54	60	0	0	0	0	0	0	54	100.0	20218	427
0.15	17	57	54	0	0	0	0	0	0	57	100.0	17642	373
0.20	8	71	49	3	3	2	1	0	0	68	95.8	16061	339
0.25	8	81	39	8	8	2	2	4	0	73	90.1	10540	225
0.30	3	91	34	10	10	5	3	2	0	81	89.0	9119	193
0.35	0	105	23	14	14	0	7	7	0	91	86.7	6350	135
0.40	0	114	14	27	27	0	21	3	3	87	76.3	2941	63
0.45	0	122	6	37	37	0	33	4	0	85	69.7	1318	27
0.50	0	128	0	69	87	0	33	36	0	59	46.1	798	15

When τ value is 0.05 in $A_{PBU\breve{G}}$, six alternatives are probabilistically assigned to their correct categories. When τ value is 0.10 or 0.15, less assignment information is obtained from the DM while all probabilistically assigned alternatives are correctly assigned to their categories. When τ is 0.20, more than half of the alternatives are probabilistically assigned while three of them are assigned to wrong categories. As the τ value increases, the number of errors increases with the number of probabilistic assignments. In case of $\tau = 0.35$, 91 of the 105 alternatives assigned probabilistic were assigned to their correct categories and a total of 14 incorrect assignments were made by assigning alternatives - that are actually in the second and third categories - to the first and fourth categories, respectively. Since $A_{PBU\breve{G}}$ calculates the probability that the utility of an alternative is greater than the category thresholds, the probability value of an alternative in at least one category is greater than 0.50. For this reason, when $\tau = 0.50$, all alternatives are probabilistically classified without asking the DM.

According to the probability values, 79 alternatives are assigned to the first category while the remaining 49 alternatives are assigned to the fourth category. This resulted in misclassification of all alternatives in the second and third categories. Moreover, the misclassified alternatives deviate exactly one category for τ values lower than 0.50. When $\tau = 0.50$, the number of CMs is higher than the number of errors indicating that there are alternatives that deviate more than one category in misclassifications.

Table 14. Performance of A_{PRENTM} in ETI problem

τ	Assignments			# of errors	# of CMs	Dist. of errors				# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM			C_1	C_2	C_3	C_4				
Non-prob.	67	-	61	-	-	-	-	-	-	-	-	15369	289
0.05	21	64	43	2	2	1	0	1	0	62	96.9	13860	285
0.10	14	74	40	3	3	0	1	0	2	71	95.9	13050	247
0.15	18	69	41	4	4	0	1	1	2	65	94.2	13158	258
0.20	12	76	40	4	4	0	0	3	1	72	94.7	13061	252
0.25	7	93	28	5	5	2	2	1	0	88	94.6	10580	202
0.30	6	94	28	6	6	1	3	1	1	88	93.6	10404	198
0.35	6	104	18	9	9	0	5	3	1	95	91.3	9782	185
0.40	6	100	22	4	4	3	0	1	0	96	96.0	9621	182
0.45	6	99	23	5	5	1	0	4	0	94	94.9	9337	173
0.50	5	108	15	10	10	1	2	7	0	98	90.7	8904	165

The assignment performance of our probabilistic algorithm, A_{PRENTM} is shown in Table 14. When τ is 0.05, 64 alternatives are probabilistically classified while two of them are incorrectly assigned to their categories. The DM requires to make 43 assignments in A_{PRENTM} while the number of DM's assignments in $A_{PBUĞ}$ is 63. In terms of the number of DM's assignments and the probabilistic assignments, the case of $\tau = 0.05$ in A_{PRENTM} is similar to the case of $\tau = 0.25$ of $A_{PBUĞ}$ where there are eight classification errors. When τ value varies between 0.10 and 0.20, the number of DM's assignments and error rates follow a similar pattern. When τ value is 0.25 and 0.30, the assignment information obtained from the DM decreases while the number of errors is limited to five and six. When τ value is 0.40 and 0.45, the number of errors decreases with an increase in the assignment information obtained from the DM. In the case of $\tau = 0.50$ where the algorithm makes the highest number of probabilistic assignments, all alternatives are probabilistically assigned in the 15th iteration when

ARE value falls below 0.5 for the first time. In terms of the number of CMs, each alternative deviates exactly one category in misclassifications for all τ values in A_{PRENTM}.

Table 15. Performance of A_{PRENTS} in ETI problem

τ	Assignments			# of errors	# of CMs	Dist. of errors				# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM			C_1	C_2	C_3	C_4				
Non-prob.	67	-	61	-	-	-	-	-	-	-	-	3656	6520
0.05	40	49	39	2	2	1	0	1	0	47	95.9	2604	406
0.10	38	52	38	3	3	1	0	2	0	49	94.2	2568	323
0.15	33	60	35	4	4	2	0	2	0	56	93.3	2438	193
0.20	30	65	33	7	7	2	0	5	0	58	89.2	2372	122
0.25	30	65	33	8	8	2	0	6	0	57	87.7	2370	110
0.30	30	67	31	9	9	3	0	6	0	58	86.6	2317	87
0.35	30	70	28	9	9	3	0	6	0	61	87.1	2227	67
0.40	30	74	24	11	11	4	0	7	0	63	85.1	2118	46
0.45	30	75	23	11	11	4	0	7	0	64	85.3	2092	44
0.50	30	77	21	12	12	4	0	8	0	65	84.4	2067	42

The number of misclassifications in A_{PRENTS} is similar to that of A_{PRENTM} for low τ values. ARE falls below 0.5 for the first time in 21st iteration in A_{PRENTS} whereas it is the 15th iteration in A_{PRENTM} when ARE value falls below 0.5 for the first time. Hence, the number of assignments of the models in A_{PRENTS} is higher than A_{PRENTM} while the number of probabilistic assignments is lower in A_{PRENTS}. The number of DM's assignments stay at a certain level due to the probabilistic assignments, hence, there is an enormous decrease in the CPU times in A_{PRENTS} when compared to the non-probabilistic case. Lastly, A_{PRENTS} correctly assign the all alternatives in the second and fourth categories in all τ values. The most common misclassification is made among the alternatives in the third category as in A_{PRENTM}. As in A_{PRENTM}, the number of CMs is equal to the number of errors for all τ values in A_{PRENTS}.

4.3.2.2 Probabilistic assignments without any τ value in ETI problem

In probabilistic algorithms, probabilistic assignments are made when the probability values are higher than a certain τ value. If the DM wants to proceed with providing the least amount of assignment information, then one can choose a τ value close to 0.5. In another case, the DM may request assignments with minimum number of classification errors and in this case a τ value close to zero can be selected. In some sorting problems, the DM may not determine any τ value. In such a case, we suggest to use relative entropy measure to select the alternative/s that will be probabilistically classified. When the ARE falls below 0.5, the alternative/s with the lowest relative entropy are probabilistically assigned to the categories regardless of the probability values. Here, we aim to assign the alternatives with the lowest assignment uncertainty.

Table 16. Performances of A_{PRENTM} and A_{PRENTS} without τ value in ETI problem

Alg.	Assignments			# of errors	Dist. of errors				# of correct assign.	% of correct assign.	# of models solved	CPU time (in sec.)
	mod.	prob.	DM		C_1	C_2	C_3	C_4				
A_{PRENTM}	19	65	44	1	1	0	0	0	64	98.5	14423	282
A_{PRENTS}	38	43	47	0	0	0	0	0	43	100.0	2865	1648

The proposed method without τ value has been applied by A_{PRENTM} and A_{PRENTS} in the ETI problem. Table 16 shows the assignment performances of the two algorithms. It can be seen that the two algorithms yield similar assignment performances with their corresponding cases where $\tau = 0.05$. A_{PRENTS} assigns all alternatives in their correct categories whereas there is one misclassification in A_{PRENTM}. Of the 65 alternatives assigned probabilistically in A_{PRENTM} without τ value, 63 of them were hypothetically assigned to only one category resulting in zero relative entropy value and an assignment probability of one. The assignment probabilities of the other two alternatives are 0.98 and 0.99 in A_{PRENTM}. A_{PRENTS}, on the other hand, makes probabilistic assignments when the assignment probability of a category is at least 0.9989. We note that the number of DM's assignments in A_{PRENTS} is higher than that of the algorithm with τ values. Hence, the CPU time of A_{PRENTS} without τ value is higher than the cases where there are τ values.

4.4 Applications on randomly generated problems

We solve additional problems to make an extensive comparison on the assignment performances of the non-probabilistic and probabilistic algorithms. We first explain the random data generation method in Section 4.4.1. We then apply the algorithms for piecewise linear and general monotone preference functions in the following sections.

4.4.1 Data generation

We generate random datasets where the scores are drawn from a Weibull distribution which is commonly used to measure product reliability and model failure time. The probability density function of the Weibull distribution is shown in (4.2) where α is the scale parameter, β is the shape parameter and x represents the failure time. The Weibull distribution can represent a variety of well-known distributions; it reduces to exponential distribution when $\beta = 1$ whereas it converges to a skewed normal distribution when $\beta = 2$. The skewness disappears for higher values of β.

$$We(x) = \alpha\beta x^{\beta-1}e^{-\alpha x^{\beta}} \tag{4.2}$$

In this study, we intend to apply the non-probabilistic and probabilistic algorithms on random data that has different distributions for each criterion. Thus, we use $\alpha = 1$ while β values are generated from a uniform distribution with $U[1,3]$. We generate independently distributed series for each criterion and then the scores are rescaled between 0 and 1. (4.3) indicates the transformed formulae to generate random number, y, that follows a Weibull distribution where x is uniformly distributed with $U[0,1]$.

$$y = \left[\frac{-1}{\alpha}\ln(1-x)\right]^{\frac{1}{\beta}} \tag{4.3}$$

In the previous applications, we have considered problems where the alternatives are assigned to three or four categories. In randomly generated problems, we study 100 alternatives to be assigned to five categories. We generate 100 random datasets to be applied in both piecewise linear and general monotone preference function cases.

4.4.2 Applications in randomly generated problems - piecewise linear case

As applications on randomly generated problems for piecewise linear preference function case, we study three criteria and three to five subintervals to evaluate 100 alternatives. We designed different preference structures for each criterion to show that our algorithms can handle various forms. We randomly define the number of subintervals separately for each criterion in each sample. For example, the first criterion may have four subintervals while the second criterion may have five subintervals in a sample problem. We then randomly generate the w_{ip} (the utility of the subinterval p in criterion g_i) and $g_i^{r_{ji}}$ (the score of the breakpoint that define the subinterval r_{ji}) values for each criterion and subinterval. We calculate the aggregate utility of the alternatives by linear interpolation. The utility thresholds for categories are randomly defined for each problem set. In order to prevent extreme imbalance on the number of alternatives to be assigned to each category, we adjust the category thresholds in a way that the number of alternatives in a category is within the range of [10, 30] for 5-category and 100-alternative problem. In constrained problems, the DM initially assigns two alternatives to each category and hence 10 alternatives are initially assigned by the DM.

4.4.2.1 Applications of the non-probabilistic algorithms in randomly generated problems - piecewise linear case

We report the averages for 100 random problems in Tables 17 and 18 for non-probabilistic case assuming a piecewise linear preference function. Table 17 shows the average assignment performances of the algorithms when there is not any constraint imposed to the problem. Table 18 indicates the average assignment performances of the non-probabilistic algorithms for constrained problem assuming piecewise linear preference function. As in the previous problems, $A_{\text{ÖZP}}$ and A_{RAND} are the worst performing ones among the all algorithms. The rest four algorithms ask similar number of questions to the DM in both unconstrained and constrained problems. Our algorithms A_{RENTM} and A_{RENTS} require the DM to assign alternatives with narrower category ranges. This is more prominent in A_{RENTM} in both unconstrained and constrained problems. This can also be observed from the numbers of PCDs. The CPU times are in parallel with the number of models solved for each algorithm except A_{RENTS}. The reason for this is the rejection rates of random weight generator in A_{RENTS}. The CPU time of A_{RENTS} decreases when the problem turns to constrained case. However, the decline is limited since the number of questions asked the DM can raise up to 60 in some problems.

Table 17. Performances of the algorithms for unconstrained non-probabilistic case in random data – piecewise linear case

Alg.	Assign. by the models	DM's assignments				Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories	Among 5 categories			
A_{RAND}	35.8	39.5	12.4	5.6	6.7	107.9	13558	246
$A_{ÖZP}$	25.6	23.8	12.0	10.5	28.1	191.7	21876	391
A_{REG}	49.3	29.2	8.1	4.1	9.3	94.9	12197	236
$A_{BUĞ}$	49.2	22.7	11.0	6.3	10.8	106.8	19017	355
A_{RENTM}	48.0	36.8	9.1	2.4	3.7	77.0	14034	253
A_{RENTS}	48.6	32.7	6.4	6.7	5.6	88.0	2622	3052

Table 18. Performances of the algorithms for constrained non-probabilistic case in random data – piecewise linear case

Alg.	Assign. by the models	DM's assignments				Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories	Among 5 categories			
A_{RAND}	47.4	19.9	13.4	7.3	12.0	116.6	4568	627
$A_{ÖZP}$	52.1	11.5	8.3	7.6	20.5	132.9	3916	597
A_{REG}	62.8	11.0	7.4	6.5	12.3	94.5	3562	556
$A_{BUĞ}$	61.4	10.6	8.2	6.1	13.7	100.1	6254	762
A_{RENTM}	62.6	17.2	5.8	3.5	10.9	82.9	3778	569
A_{RENTS}	63.5	13.7	6.1	4.9	11.8	87.8	1252	1806

4.4.2.2 Applications of the probabilistic algorithms in randomly generated problems - piecewise linear case

We apply the probabilistic algorithms in randomly generated problems assuming piecewise linear preference function. Tables 19, 20 and 21 show the average assignment performances of the probabilistic algorithms $A_{PBU\check{G}}$, A_{PRENTM} and A_{PRENTS} for different τ values, respectively. In order to compare the probabilistic assignments with non-probabilistic case, the assignment performance of the non-probabilistic case is initially reported in each algorithm.

Table 19. Performance of $A_{PBU\check{G}}$ in random data – piecewise linear case

τ	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
Non-prob.	49.2	-	50.8	-	-	-	-	-	-	-	-	19017	355
0.05	26.8	26.7	46.5	1.8	0.0	0.0	0.5	0.4	0.9	24.9	93.3	17440	314
0.10	14.4	42.9	42.7	5.2	0.9	0.1	0.6	0.5	3.1	37.7	87.9	15220	280
0.15	7.8	55.4	36.8	9.0	2.4	0.9	1.0	1.0	3.8	46.4	83.8	12300	217
0.20	4.0	65.4	30.6	16.1	3.2	2.9	1.6	4.0	4.4	49.3	75.4	9374	135
0.25	1.9	73.7	24.4	25.1	4.8	6.5	4.7	5.7	3.4	48.6	65.9	6973	89
0.30	1.6	81.6	16.8	36.5	7.2	9.5	7.6	7.8	4.4	45.1	55.3	4618	56
0.35	0.5	89.5	10.0	48.9	10.6	12.8	10.1	11.1	4.3	40.6	45.4	2899	44
0.40	0.0	93.6	6.4	59.3	11.0	15.6	14.5	14.4	3.9	34.3	36.6	1833	26
0.45	0.0	96.6	3.4	69.2	11.1	17.0	19.2	17.6	4.3	27.4	28.4	1189	20
0.50	0.0	100	0.0	75.6	11.6	18.1	22.1	18.6	5.1	24.4	24.4	833	13

Table 20. Performance of A_{PRENTM} in random data – piecewise linear case

τ	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
Non-prob.	48.0	-	52.0	-	-	-	-	-	-	-	-	14034	253
0.05	20.4	39.1	40.5	2.7	0.1	1.2	0.3	1.0	0.1	36.4	93.1	12941	246
0.10	18.0	44.8	37.2	4.1	0.7	1.3	0.7	1.1	0.3	40.7	90.8	12530	240
0.15	17.0	47.8	35.2	5.5	0.9	1.6	1.0	1.6	0.4	42.3	88.5	12262	232
0.20	16.9	50.9	32.2	7.7	1.1	3.2	1.3	1.7	0.4	43.2	84.9	12049	225
0.25	17.0	51.8	31.2	9.5	1.3	2.9	1.4	3.2	0.7	42.3	81.7	11923	219
0.30	16.5	53.0	30.5	10.6	1.3	3.5	1.6	3.7	0.3	42.4	80.0	11796	214
0.35	15.4	57.0	27.6	12.5	1.6	4.4	1.3	5.0	0.2	44.5	78.1	11575	210
0.40	15.8	56.9	27.3	13.5	1.1	4.4	2.0	5.7	0.3	43.4	76.3	11520	201
0.45	15.3	59.5	25.2	14.8	1.3	5.0	2.1	5.2	1.2	44.7	75.1	11380	196
0.50	15.1	60.9	24.0	16.0	1.4	5.1	2.4	6.0	1.1	44.9	73.7	11306	186

In general, our algorithms, A_{PRENTM} and A_{PRENTS} perform better than $A_{PBUĞ}$ in terms of the correct assignment of the alternatives. Our algorithms make higher number of probabilistic assignments than $A_{PBUĞ}$ for low τ values while the misclassification rates are similar for the three algorithms. $A_{PBUĞ}$ probabilistically assign all alternatives when τ increases to 0.5 without eliciting any assignment information from the DM as in ETI problem. Hence, the number of misclassifications is high in $A_{PBUĞ}$ for high τ values. The classification errors mostly occurr in the second, third and fourth categories in all algorithms. The CPU time to complete the task in A_{PRENTS} is very low when compared to the non-probabilistic case since the number of assignment information obtained from the DM is low in probabilistic case.

Table 21. Performance of A_{PRENTS} in random data – piecewise linear case

τ	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
Non-prob.	48.6	-	51.4	-	-	-	-	-	-	-	-	2622	3052
0.05	18.4	40.9	40.7	1.0	0.1	0.4	0.1	0.4	0.0	39.9	97.6	2175	213
0.10	17.8	45.1	37.1	2.0	0.3	0.7	0.2	0.8	0.0	43.1	95.6	2069	142
0.15	19	45.5	35.5	3.5	0.7	1.0	0.3	1.3	0.2	42.0	92.3	2023	161
0.20	16.7	49.1	34.2	4.5	0.6	1.0	0.7	2.0	0.2	44.6	90.8	1980	100
0.25	16.2	51.6	32.2	6.4	0.9	1.7	1.2	2.3	0.3	45.2	87.6	1920	73
0.30	15.6	54.5	29.9	8.0	1.2	2.3	1.7	2.5	0.3	46.5	85.3	1845	63
0.35	14.5	57.0	28.5	10.0	1.3	2.6	2.6	3.1	0.4	47.0	82.5	1807	48
0.40	13.5	59.1	27.4	11.4	1.4	2.7	3.6	3.3	0.4	47.7	80.7	1773	42
0.45	14	60.5	25.5	13.1	1.8	2.8	4.5	3.6	0.4	47.4	78.3	1719	38
0.50	13.5	62.7	23.8	15.0	1.8	3.2	5.4	4.1	0.5	47.7	76.1	1671	33

When the DM does not provide any τ value to define the probabilistic assignments, we probabilistically assign the alternative/s with the lowest relative entropy when the ARE falls below 0.5. The reason for choosing the alternatives with lowest relative entropy is that such alternatives are expected to be hypothetically assigned to one category in most cases. Table 22 shows the assignment performances of our algorithms, A_{PRENTM} and A_{PRENTS} when there is no τ requirement for defining the probabilistic assignments. A_{PRENTM} makes higher number of probabilistic assignments than A_{PRENTS} while A_{PRENTS} require the DM to assign higher number of alternatives than A_{PRENTM}. The number of misclassifications in two algorithms are less than the cases where $\tau = 0.05$.

Table 22. Performances of A_{PRENTM} and A_{PRENTS} without τ value in random data

Alg.	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
A_{PRENTM}	29.3	28.1	42.6	1.7	0.2	0.6	0.4	0.4	0.1	26.4	94.0	13366	214
A_{PRENTS}	28.1	25.0	46.9	0.4	0.0	0.3	0.0	0.1	0.0	24.6	98.4	2510	1193

4.4.3 Applications on randomly generated problems – general monotone case

We further apply the proposed and benchmark algorithms on randomly generated problems assuming general monotone preference function. In general monotone preference function case, we generate the same 100 random datasets as in the piecewise linear case. Recall that x_1^i, $x_2^i, \ldots,$ $x_{m_i}^i$ are the ordered score values in criterion g_i in general monotone preference function case. Hence, we randomly generate the marginal utility of m_i criteria scores for each criterion g_i to represent the preference structure of the DM. By this way, we calculate the aggregate utilities of the alternatives. The category thresholds are defined by the same way explained in piecewise linear case. We apply the non-probabilistic and probabilistic algorithms in randomly generated problems in Section 4.4.3.1 and 4.4.3.2, respectively.

4.4.3.1 Applications of the non-probabilistic algorithms in randomly generated problems – general monotone case

Tables 23 and 24 show the average assignment performance of the proposed non-probabilistic algorithm, A_{RENTM}, and the benchmark algorithms for unconstrained and constrained problems. In general, the cognitive burden of the DM is higher in each algorithm for general monotone case when compared to the piecewise linear case. In line with the findings in the previous problems, A_{RAND} and $A_{\ddot{O}ZP}$ are the worst performing algorithms in randomly generated problems.

As in the bus revision problem where the preferences of the DM is assumed to be consistent with a general monotone function, $A_{BUĞ}$ performs worse than A_{REG} and A_{RENTM} in terms of the assignments by the models and the number of models solved. Our algorithm, A_{RENTM}, require the DM to assign alternatives with narrower category ranges when compared to A_{REG} although the two algorithms ask the same number of questions to the DM while solving similar number of models.

Table 23. Performances of the algorithms for unconstrained non-probabilistic case in random data – general monotone case

Alg.	Assign. by the models	DM's assignments				Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories	Among 5 categories			
A_{RAND}	12.2	30.2	17.8	19.6	20.2	305.4	27450	553
$A_{ÖZP}$	17.7	15.3	16.8	18.2	32.0	331.5	27250	520
A_{REG}	29.1	20.8	19.9	15.3	14.9	266.1	18741	401
$A_{BUĞ}$	21.0	20.2	21.2	22.1	15.5	290.9	34405	563
A_{RENTM}	29.3	23.1	20.8	17.6	9.2	252.3	19778	458

Table 24. Performances of the algorithms for constrained non-probabilistic case in random data – general monotone case

Alg.	Assign. by the models	DM's assignments				Number of PCDs	Number of models solved	CPU time (in seconds)
		Among 2 categories	Among 3 categories	Among 4 categories	Among 5 categories			
A_{RAND}	19.8	27.8	23.2	16.9	12.3	174.1	9248	956
$A_{ÖZP}$	28.3	19.3	9.4	14.5	28.5	195.6	6723	689
A_{REG}	43.1	23.7	18.2	7.9	7.1	112.2	5473	744
$A_{BUĞ}$	35.9	25.7	20.6	9.9	7.9	128.2	9852	983
A_{RENTM}	44.5	26.8	15.3	8.7	4.7	102.3	5324	762

4.4.3.2 Applications of the probabilistic algorithms in randomly generated problems – general monotone case

We apply the probabilistic algorithms, $A_{PBUĞ}$ and A_{PRENTM}, in randomly generated problems and report the average values in Tables 25 and 26 for $A_{PBUĞ}$ and A_{PRENTM}, respectively. In general, the misclassification rates are higher in general monotone case when compared to the four-category ETI problem where the preferences of the DM is assumed to be consistent with piecewise linear function. The numbers of probabilistic assignments and misclassifications are lower in $A_{PBUĞ}$ for low τ values. When τ increases to 0.15, the two algorithms make similar number of misclassifications while A_{PRENTM} makes higher number of probabilistic assignments. The number of probabilistic assignments is higher in $A_{PBUĞ}$ for higher τ values. However, 70% of the probabilistic assignments are recorded as misclassifications.

Table 25. Performance of $A_{PBUĞ}$ in random data – general monotone case

τ	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
Non-prob.	21.0	-	79.0	-	-	-	-	-	-	-	-	34405	563
0.05	9.2	10.4	80.4	3.8	0.0	0.4	1.0	2.5	0.0	6.6	63.5	26035	454
0.10	7.9	21.5	70.6	10.5	0.1	1.1	3.1	6.2	0.0	11.0	51.2	20554	350
0.15	7.0	32.9	60.1	19.5	0.6	2.3	5.8	10.4	0.3	13.4	40.7	13996	252
0.20	5.8	44.6	49.6	29.3	1.8	4.8	9.4	12.7	0.6	15.3	34.3	10463	178
0.25	4.8	54.3	40.9	37.4	2.9	7.2	12.6	13.3	1.4	16.9	31.1	8063	129
0.30	3.5	64.0	32.5	45.1	4.5	10.2	14.9	13.5	2.0	18.9	29.5	4985	88
0.35	1.8	74.7	23.5	53.1	7.0	13.3	16.3	14.1	2.3	21.6	28.9	3975	53
0.40	0.5	83.4	16.1	59.3	9.3	16.3	16.3	14.5	2.9	24.1	28.9	1864	31
0.45	0.0	90.8	9.2	64.5	11.3	17.8	17.0	14.9	3.5	26.3	29.0	1240	18
0.50	0.0	100.0	0.0	71.1	13.2	19.2	17.5	16.1	5.1	28.9	28.9	814	11

Table 26. Performance of A_{PRENTM} in random data – general monotone case

τ	Assignments			# of errors	Dist. of errors					# of correct assign.	% of correct assign.	# of models solved	CPU time (in seconds)
	models	prob.	DM		C_1	C_2	C_3	C_4	C_5				
Non-prob.	29.3	-	70.7	-	-	-	-	-	-	-	-	19788	458
0.05	16.5	14.6	68.9	7.3	2.0	1.5	1.3	0.9	1.6	7.3	50.0	21528	435
0.10	12.4	23.6	64	12.0	0.6	5.4	3.7	1.9	0.4	11.6	49.2	20884	423
0.15	11.6	30.9	57.5	17.9	1.1	6.7	5.9	3.0	1.2	13	42.1	19993	402
0.20	11.1	36.5	52.4	20.8	1.2	8.0	6.9	3.6	1.1	15.7	43.0	19192	376
0.25	10.9	39.5	49.6	24.1	2.6	8.1	7.2	5.5	0.6	15.4	39.0	18652	364
0.30	10	46.5	43.5	28.9	2.7	10.2	8.0	6.3	1.5	17.6	37.8	17742	355
0.35	9.9	50.5	39.6	31.2	2.8	10.7	8.9	6.0	2.7	19.3	38.2	17203	333
0.40	10.1	52.0	37.9	31.7	3.6	10.4	8.9	5.5	3.3	20.3	39.0	16991	329
0.45	9.9	52.8	37.3	32.2	3.6	10.7	8.8	5.6	3.5	20.6	39.0	16933	326
0.50	10	53.3	36.7	32.8	3.9	10.3	9.2	6.3	3.1	20.5	38.5	16793	325

Our algorithm, A_{PRENTM}, makes higher number of correct assignments for lower τ values. When $\tau \geq 0.35$ in A_{PRENTM}, the number of misclassifications does not increase at a significant rate as in $A_{PBU\check{G}}$ since the ARE rule prevents the probabilistic assignments in case of insufficient number of assignments made by the DM. Although the number of models decreases with an increasing τ value, it remains at a certain level due to the ARE rule in A_{PRENTM}. The assignment performance of no τ case is shown in Table 26. The number of misclassifications in A_{PRENTM} is less than the probabilistic cases while the number of information obtained from the DM is higher in no τ case.

4.5 Stopping condition in non-probabilistic case

The proposed probabilistic algorithms lead to decrease the assessment burden of the DM especially for higher τ values while the misclassifications are at acceptable level. However, as mentioned before, the cognitive burden of the DM increases with the increase in the number of categories, criteria or subintervals in non-probabilistic case. For instance, in randomly generated problems with 5 categories, 3 criteria and 100 alternatives, almost 70 assignments are needed to be made by the DM so that the model helps sorting approximately 30 alternatives. In this section we apply the stopping condition such that the non-probabilistic algorithms will terminate once the DM assigns a certain number of alternatives. As an application of non-probabilistic algorithms, we study the unconstrained case of MBA problem where 81 alternatives evaluated on three criteria and three subintervals are assigned to three categories. We consider the DM's assignments of 10, 20, and 30 alternatives and check the number of assignments by the models.

Table 27 shows the assignment performances of the non-probabilistic algorithms under this stopping condition. When the DM assigns 10 alternatives, the algorithms assign fewer number of alternatives. Recall that the dominance relations help to narrow down the category ranges in the first iterations. The best algorithm is surprisingly random algorithm when the DM assigns 10 alternatives. In terms of the number of PCDs, our algorithms perform better than the benchmark algorithms in all

10, 20 and 30-alternative cases. Our algorithms less frequently consult the DM in 20 and 30-alternative cases. When the DM assigns 30 alternatives, approximately 87% of the alternatives are assigned to their true categories. Hence, the stopping condition can be helpful in such an interactive sorting process.

Table 27. Performances of non-probabilistic algorithms – stopping condition case

Alg.	DM's assign.	Assign. by the models	# of PCDs	# of models solved	CPU time (in seconds)
A_{RAND}	10	5.7	16.2	2645	44
$A_{ÖZP}$	10	0	20	3344	46
A_{REG}	10	2.8	16.7	2685	41
$A_{BUĞ}$	10	4	14	4951	64
A_{RENTM}	10	3	13	2256	36
A_{RENTS}	10	1	13	425	18
A_{RAND}	20	9.7	28.6	4116	71
$A_{ÖZP}$	20	0	40	5964	83
A_{REG}	20	12.7	27.6	4097	64
$A_{BUĞ}$	20	12	26	7148	103
A_{RENTM}	20	16	23	3318	58
A_{RENTS}	20	22	23	774	39
A_{RAND}	30	15.4	39.5	5094	92
$A_{ÖZP}$	30	0	60	7616	110
A_{REG}	30	33.8	37.7	4719	73
$A_{BUĞ}$	30	34	36	7958	117
A_{RENTM}	30	41	33	3741	66
A_{RENTS}	30	40	33	1062	128

4.6 Comparison of the performances of algorithms

In this section, we present the performance metrics of the proposed and benchmark algoritms. We first give the comparisons of the non-probabilistic algorithms applied on MBA and bus revision problem through radar charts in Section 4.6.1 and then present the performances of the probabilistic algorithms applied on ETI and randomly generated problems in Section 4.6.2.

4.6.1 Comparison of non-probabilistic algorithms – MBA and bus revision problems

In our experiments on non-probabilistic algorithms, two different problems - MBA problem and bus revision problem - with and without constraints are addressed. In the first problem where there are three categories, three criteria, and 81 alternatives, the preferences of the DM are assumed to be consistent with an underlying piecewise linear utility function. On the other hand, in the second problem where there are four categories, eight criteria and 76 alternatives, a general monotone preference function is assumed to represent the preferences of the DM.

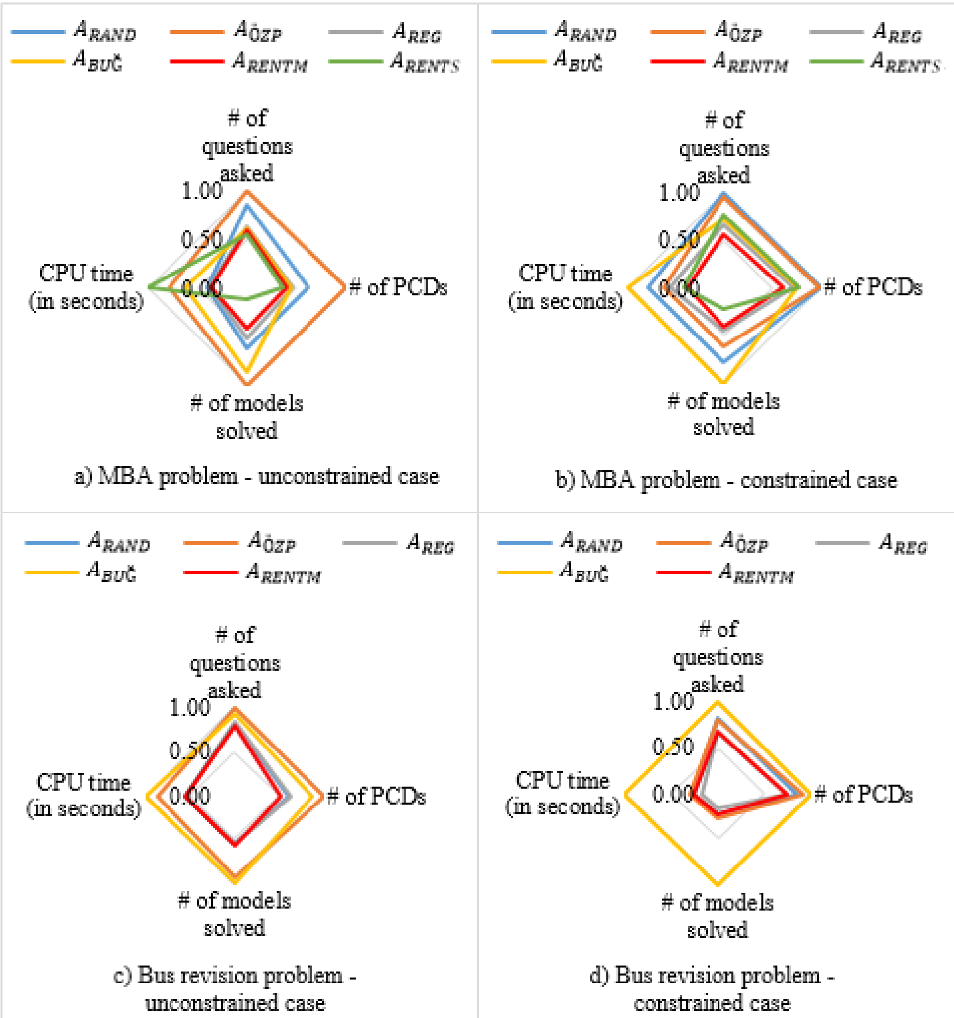


Figure 16. Overall performances of the non-probabilistic algorithms on MBA and bus revision problem

The overall performances of the proposed and benchmark non-probabilistic algorithms are summarized in Figure 16 with respect to four performance metrics: (i) the number of questions asked, (ii) the number of PCDs, (iii) the number of models solved, and (iv) CPU time (in seconds). The performances of the non-probabilistic algorithms are given for unconstrained and constrained cases of MBA and bus revision problems separately. In each case, the performance metrics are scaled to one in order to establish the charts. Our simulation-based algorithm, A_{RENTS}, performs well in terms of the information obtained from the DM in MBA problem in both unconstrained and constrained cases in comparison with the benchmark algorithms. Moreover, A_{RENTS} solves the lowest number of models in each case. However, A_{RENTS} has higher CPU time in unconstrained case due to excessive time for generating 10,000 sets of decision weights. In constrained case, A_{RENTS} has lower CPU time than the benchmark algorithms thanks to the lower number of iterations and models solved. Our model-based algorithm, A_{RENTM}, on the other hand, requires less amount of information from the DM in comparison with the benchmark algorithms in both MBA problem and bus revision problem. Furthermore, it takes less amount of time to complete the assignment process when compared to A_{RENTS} in MBA problem and benchmark algorithms in each case.

4.6.2 Comparison of probabilistic algorithms – ETI and randomly generated problems

In order to measure the assignment performances of the probabilistic algorithms, we apply the proposed and benchmark algorithms on ETI problem and several randomly generated problems. In ETI problem where there are four categories, four criteria, and 128 alternatives, the preferences of the DM are assumed to be consistent with an underlying piecewise linear utility function. In randomly generated problems, we generate five-category, three-criteria and 100-alternative datasets based on Weibull distribution. For comparison, we present general monotone preference function case in randomly generated problems. We further give the performances of non-probabilistic algorithms in each problem.

The overall performances of the proposed and benchmark algorithms in 128-alternative ETI problem are summarized in Table 28 with respect to four performance metrics: (i) the number of probabilistic assignments, (ii) the number of DM's assignments, (iii) the number of errors, and (iv) CPU time (in seconds). The performances of the probabilistic algorithms are summarized for different τ values. The results show that the algorithm of Buğdacı et al. (2013), $A_{PBUĞ}$, assigns fewer number of probabilistic assignments and hence asks the DM to assign higher number of alternatives for lower τ values. Moreover, $A_{PBUĞ}$ makes higher number of misclassifications for higher τ values. Our model-based algorithm, A_{PRENTM} makes higher number of probabilistic assignments than our simulation-based algorithm A_{PRENTS} while the two algorithms have similar number of misclassifications. In terms of the CPU time, A_{PRENTM} has shorter CPU time for lower τ values while A_{PRENTM} takes less CPU time for higher τ values due to the decline in the number of DM's assignments.

Table 28. Overall performances of the probabilistic algorithms in ETI problem

τ	Assignments						% of correct assign.			CPU time (in seconds)		
	probabilistic			DM								
	$A_{PBUĞ}$	A_{PRENTM}	A_{PRENTS}	$A_{PBUĞ}$	A_{PRENTM}	A_{PRENTS}	$A_{PBUĞ}$	A_{PRENTM}	A_{PRENTS}	$A_{PBUĞ}$	A_{PRENTM}	A_{PRENTS}
Non-prob.	-	-	-	69	61	61	-	-	-	514	289	6520
0.05	36	64	49	63	43	39	100	96.9	95.9	465	285	406
0.2	71	76	65	49	40	33	95.8	94.7	89.2	339	252	122
0.35	105	104	70	23	18	28	86.7	91.3	87.1	135	185	67
0.5	128	108	77	0	15	21	46.1	90.7	84.4	15	165	42

Table 29 shows the assignment performances of the $A_{PBUĞ}$ and A_{PRENTM} in general monotone case of randomly generated problems where 100 alternatives are assigned to five categories. The results are similar to the ETI problem in terms of the number of probabilistic and DM's assignments as well as the number of the number of misclassifications. The number of DM's assignments and the number of misclassifications are higher when compared to the ETI problem although the number of alternatives evaluated is higher in ETI problem. One possible reason for this can be the characteristics of the piecewise linear and general monotone preference functions.

Table 29. Overall performances of the probabilistic algorithms in randomly generated problems – general monotone case

τ	Assignments				% of correct assign.		CPU time (in seconds)	
	probabilistic		DM					
	$A_{PBU\breve{G}}$	A_{PRENTM}	$A_{PBU\breve{G}}$	A_{PRENTM}	$A_{PBU\breve{G}}$	A_{PRENTM}	$A_{PBU\breve{G}}$	A_{PRENTM}
Non-prob.	-	-	79	70.7	-	-	563	458
0.05	10.4	23.6	80.4	64	63.5	49.2	454	435
0.2	44.6	39.5	49.6	49.6	34.3	39.0	178	376
0.35	74.7	52.0	23.5	37.9	28.9	39.0	53	333
0.5	100.0	53.7	0.0	36.4	28.9	38.4	11	325

CHAPTER 5

CONCLUSIONS

In this study, we develop interactive algorithms to assign alternatives into pre-defined number of ordered categories. We assume that the preferences of DM are consistent with an additive function in piecewise linear or general monotone form. We consider the cases where the DM may or may not provide the size of the categories and initial assignment information. We define the possible categories the alternatives can be assigned throughout the LP or MIP models. We hypothetically assign the unlabeled alternatives into categories based on a set of parameters compatible with the preferences of the DM.

We suggest two approaches to utilize a set of compatible parameters that are used to make hypothetical assignments. In model-based approach, we follow an ad hoc procedure by taking the average of the minimum and maximum scores of category thresholds as well as the utility scores of the unlabeled alternatives derived from mathematical models to hypothetically assign the unlabeled alternatives. In simulation-based approach, we generate 10,000 random sample sets with Monte Carlo simulations assuming uniform distributions for parameters. We eliminate the sets of parameters that are not compatible with the existing assignments of the DM.

Based on several hypothetical assignments, we estimate the probability of belonging to a category for each alternative. Using the relative entropy measure, we find the most ambiguous alternative to ask the DM for assignment. The relative entropy method enables us to make comparison between alternatives with different category ranges in terms of the uncertainty levels to belong to a category. We update the probabilities iteratively utilizing the gathered information on the assignments of alternatives by the DM until all alternatives are assigned to the categories. Our non-probabilistic algorithm guarantees assigning alternatives to their exact categories assuming that the preferences of the DM are consistent with an additive preference function. The proposed probabilistic algorithm on the other hand, allows the alternatives to be assigned with respect to the assignments probabilities when sufficient assignment information is obtained from the DM.

We demonstrate the performances of our probabilistic and non-probabilistic algorithms against a state-of-the-art algorithm on three problems from the literature and several randomly generated problems. We add three alternative selection approaches from the literature that are employed within our algorithm in non-probabilistic case in order to measure the performance of the alternative selection approach based on the relative entropy. We consider the cases with/without category size restrictions and initial assignments in non-probabilistic problem setting. We assume either piecewise linear or general monotone preference function in three problems from the literature and we consider the two functional forms in randomly generated problems.

The results indicate that our algorithms tend to select among fewer categories when asking the DM to make assignment in non-probabilistic case. Our algorithms perform well in terms of the assignment information obtained from the DM and the number of models solved. One may expect that the assignment of alternatives with wider category ranges will bring more valuable information to the system. However, we show that the category ranges do not represent the ambiguity of assignments and a relative entropy-based method works well in identifying the ambiguity of alternatives.

We test the performances of our probabilistic algorithms against the benchmark algorithm on an energy related problem and several randomly generated problems. In general, it is seen that the rule of not assigning probabilistically without getting enough information from the DM result in low number of misclassifications when compared to the benchmark algorithm. We also consider to probabilistically assign the alternative/s with lowest relative entropy. This approach yields a similar assignment performance with the case of small τ values where the number of information obtained from the DM is high and the number of misclassifications is low.

The algorithms with model-based and simulation-based hypothetical assignments yield similar performances in terms of assignment information obtained from the DM and the accuracy of the probabilistic assignments. Simulation-based technique takes an extensive amount of time to generate 10,000 compatible set of parameters when the DM assigns sufficient number of alternatives. We show that it is not possible to conduct a simulation-based approach in an interactive setting where the preferences of the DM is assumed to be consistent with a general monotone preference function.

The models used in this study always give a feasible solution when the DM's preferences are consistent with the assumed additive structure. However, it may be possible for the DM's assignments to be inconsistent with an additive preference function. The models will detect such inconsistencies by resulting in infeasible solutions. One way to handle the inconsistencies is to present the conflicting set of assignments to the DM to make a revision about his/her preferences. In addition, our approach enables the DM to go back to earlier stages of the process as in Ciomek et al. (2017). Another way to handle the inconsistencies is to address the infeasibility of the mathematical models. There are several ways to deal with such infeasibilities such as minimizing the summation of classification errors, minimizing the maximum classification error, and minimizing the number of misclassifications (see for example Chinneck, 2008). A recent work of Kadzinski et al. (2020) considers the contingencies in DM's behavior to handle the inconsistencies in MCS problems.

Our algorithms are directly applicable to many problems such as patient or supplier classification. Categorization of the patients as emergent and non-emergent takes an important role in planning the allocation of nursing staff in accordance with the nursing care needs. The probabilistic algorithm is appropriate for patient classification since this kind of a problem can allow the misclassifications. On the other hand, there are studies such as Manshadi et al. (2015) that apply UTADIS to supplier classification. Our non-probabilistic and probabilistic algorithms can be applied to supplier classification assuming a piecewise-linear preference function.

As a future research direction, one can extend this work to develop an interactive approach to MCS problems with non-monotonic criteria. Other further research areas can be using a Bayesian approach for identifying the probabilities of assignments and working with different utility function forms such as quasiconcave or L_p norm functions. The characteristics of different functional forms can be combined with the relative entropy to help the DM in the assignment of alternatives to categories.

REFERENCES

Abbas, A. (2004). Entropy methods for adaptive utility elicitation. *IEEE Transactions on Systems Man and Cybernetics – Part A: Systems and Humans*, *34*(2), 169–178.

Arrow, K. J., and Enthoven, A. C. (1961). Quasi-Concave Programming, *Econometrica*, *29*, 779–800.

Bana e Costa, C. A., De Corte, J. M., and Vansnick, J. C. (2005). On the Mathematical Foundations of MACBETH, in Multiple Criteria Decision Analysis: State of the Art Surveys, J. Figueira, S. Greco, M. Ehrgott (eds.), Springer, New York, 409-442.

Belacel, N., and Boulassel, M. R. (2004). Multicriteria fuzzy classification procedure PROCFTN: methodology and medical application. *Fuzzy Sets and Systems*, *141*, 203–217.

Benabbou, N., Perny, P., and Viappiani, P. (2017). Incremental elicitation of Choquet capacities for multicriteria choice, ranking and sorting problems. *Artificial Intelligence*, *246*, 152–180.

Berkson, J. (1944). Application of the logistic function to bio-assay. *Journal of American Statistical Association, 39*, 357–365.

Bliss, C. (1934). The methods of probits. *Science,* 79, 38–39.

Bouyssou, D., and Marchant, T. (2007a). An axiomatic approach to noncompensatory sorting methods in MCDM, I: The case of two categories. *European Journal of Operational Research*, *178*(1), 217–245.

Bouyssou, D., and Marchant, T. (2007b). An axiomatic approach to noncompensatory sorting methods in MCDM, II: More than two categories. *European Journal of Operational Research*, *178*(1), 246–276.

Branke, J., Corrente, S., Greco, S., and Gutjahr, W. (2017). Efficient pairwise preference elicitation allowing for indifference. *Computers & Operations Research*, *88*, 175 – 186.

Brans, J. P., Vıncke, P. H., and Mareschal, B. (1986). How to select and how to rank projects: The PROMETHEE method. *European Journal of Operational Research*, 24, 228-238.

Buğdacı, A. G., Köksalan, M., Özpeynirci, S., and Serin, Y. (2013). An Interactive probabilistic approach to multi-criteria sorting. *IIE Transactions*, *45*, 1048–1058.

Charnes, A., Cooper, W. W., and Rhodes, E. (1978). Measuring Efficiency of Decision Making Units. *European Journal of Operational Research*, *2*, 429-444.

Chinneck, J. W. (2008). *Feasibility and infeasibility in optimization: algorithms and computational methods*, Springer: New York, USA.

Choquet, G. (1955). Theory of capacities. *Annales de L' Institut Fourier*, *5*, 131–295.

Ciomek, K., Kadziński, M., and Tervonen, T. (2017). Heuristics for prioritizing pair-wise elicitation questions with additive multi-attribute value models. *Omega*, *71*, 27–45.

Çelik, B., Karasakal, E., and İyigün, C. (2015). A probabilistic multiple criteria sorting approach based on distance functions. *Expert Systems with Applications*, *42*(7), 3610–3618.

Denneberg, D. (1994). *Non-additive measure and integral*. Kluwer Academic Publishers, Boston.

Devaud, J. M., Groussaud, G., and Jaquet-Lagreze, E. (1980). UTADIS: Une méthode de construction de fonctions d'utilité additives rendant compte de jugements globaux. in: Proceedings of the European working group on MCDA. Bochum, Germany.

Devinney, T., Dowling, G. R., Perm-Ajchariyawong, N. (2008). The Financial Times business school ranking: What quality is this signal of quality? *European Management Review, 5,* 195–208.

Diakoulaki, D., Zopounidis, C., Mavrotas, G., and Doumpos, M. (1999). The use of a preference disaggregation method in energy analysis and policy making. *Energy*, *24*(2), 157–166.

Dias, L. C., Mousseau, V., Figueira, J., and Climaco, J. (2002). An aggregation / disaggregation approach to obtain robust conclusions with ELECTRE TRI. *European Journal of Operational Research*, *138*(2), 332–348.

Doumpos, M., and Zopounidis, C. (2002). *Multicriteria Decision Aid Classification Methods*. AH Dordrecht, The Netherlands: Kluwer.

Doumpos, M., and Zopounidis, C. (2004). Developing sorting models using preference disaggregation analysis: An experimental investigation. *European Journal of Operational Research*, *154*(3), 585–598.

Doumpos, M., and Zopounidis, C. (2011). Preference disaggregation and statistical learning for multicriteria decision support: a review. *European Journal of Operational Research*, *209*(3), 203–214.

Doyle, J., and Green, R. (1994). Efficiency and Cross-Efficiency in DEA: Derivations, Meanings and Uses. *The Journal of the Operational Research Society*, *45*(5), 567-578.

Driver, H. E., and Kroeber, A. L. (1932). Quantitative expression of cultural relationships. *University of California Publications in Archaeology and Ethnology*, *31*, 211–216.

Edwards, W. (1971). Social utilities. *Engineering Economist*, Summer Symposium Series, 6, 119-129.

Edwards, W., and Barron, F. H. (1994). SMARTS and SMARTER: Improved simple methods for multiattribute utility measurement, *Organizational Behavior and Human Decision Processes 60,* 306-325.

Figueira, J., Greco, S., and Slowinski, R. (2009). Building a set of additive value functions representing a reference preorder and intensities of preference: GRIP method. *European Journal of Operations Research, 195*(2), 460–486.

Fisher, R. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics, 7*, 179–188.

Ghaderi, M., Ruiz, F., and Agell, N. (2017). A ming approach for learning non-monotonic additive value functions in multiple criteria decision aiding. *European Journal of Operational Research, 259*(3), 1073–1084.

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika, 53,* 325-338.

Grabisch, M. (1995). Fuzzy integral in multicriteria decision making. *Fuzzy Sets Syst. 69*(3), 279–298.

Greco, S., Matarazzo, B., and Slowinski, R. (2001). Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research, 129*(1), 1–47.

Greco, S., Matarazzo, B., and Slowinski, R. (2002). Rough sets methodology for sorting problems in presence of multiple attributes and criteria. *European Journal of Operational Research, 138*(2), 247–259.

Greco, S., Matarazzo, B., and Slowinski, R. (2004). Axiomatic characterization of a general utility function and its particular cases in terms of conjoint measurement and rough-set decision rules. *European Journal of Operational Research, 158*(2), 271–292.

Greco, S., Mousseau, V., and Słowinski, R. (2008). Ordinal regression revisited: Multiple criteria ranking with a set of additive value functions. *European Journal of Operational Research*, *191*, 415–435.

Greco, S., Mousseau, V., and Słowinski, R. (2010). Multiple criteria sorting with a set of additive value functions. *European Journal of Operational Research, 207*(3), 1455–1470.

Hillier, F. S. (2001). *Evaluation and decision models: a critical perspective*. Kluwer, Boston.

Holloway, H. A and White, C. C. (2003). Question selection for multi-attribute decision-aiding. *European Journal of Operational Research, 148*, 525–33.

Hwang, C. L., and Yoon, K. P. (1981). *Multiple Attribute Decision Making: Methods and Applications*; Springer: New York, USA.

Ishizaka, A., and Nemery, P. (2013). *Multi-criteria decision analysis: methods and software*. John Wiley and Sons, Chichester, United Kingdom.

Jacquet-Lagreze, E. (1995). An application of the UTA discriminant model for the evaluation of R&D projects, in: Pardalos PM; Siskos Y; Zopounidis C (Eds.). Advances in Multicriteria Analysis, Kluwer Academic Publishers, Dordrecht, pp. 203–211.

Jacquet-Lagreze, E., and Siskos, Y. (1982). Assessing a set of additive utility functions for multicriteria decision making: The UTA method. *European Journal of Operational Research, 10*, 151–164.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical Review, 106*, 620–630.

Kadzinski, M., Ghaderi, M., and Dabrowski, M. (2020). Contingent preference disaggregation model for multiple criteria sorting problem, *European Journal of Operations Research, 281*, 369–387.

Kadzinski, M., and Tervonen, T. (2013). Stochastic ordinal regression for multiple criteria sorting problems. *Decision Support Systems, 55*, 55–66.

Kang, T. H. A., Frej, E. A., and De Almeida, A. T. (2020). Flexible and Interactive Tradeoff Elicitation for Multicriteria Sorting Problems. *Asia Pacific Journal of Operations Research, 37*(5), 1–22.

Karakaya, G., Galelli, S., Ahipasaoglu, S., and Taormina, R. (2016). Identifying (quasi) equally informative subsets in feature selection problems for classification: a max-relevance min-redundancy approach. *IEEE Transactions on Cybernetics, 46*(6), 1424–1437.

Karsu, Ö. (2013). Using holistic multicriteria assessments: the convex cones approach. In: Cochran J. (ed) Wiley encyclopedia of operations research and management science. Wiley, New York, pp 1–14.

Karsu, Ö., Morton, A., and Argyris, N. (2018). Capturing preferences for inequality aversion in decision support. *European Journal of Operational Research, 264*(3),686–706.

Kartal, H., Oztekin, A., Gunasekaran, A., and Cebi, F. (2016). An integrated decision analytic framework of machine learning with multi-criteria decision making for multi-attribute inventory classification. *Computers & Industrial Engineering, 101*, 599–613.

Keeney, R. L., and Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value tradeoffs*. New York: Wiley.

Kim, B., Gel, E. S., Fowler, J. W., Carlyle, W. M., and Wallenius, J. (2006). Evaluation of nondominated solution sets for k-objective optimization problems: an exact method and approximations. *European Journal of Operational Research, 173*, 2, 565–582.

Korhonen, P., Moskowitz, H., and Wallenius, J. (1992). Multiple criteria decision support - A review. *European Journal of Operational Research, 63*(3), 361–375.

Korhonen P., and Soismaa M. (1981). An interactive multiple criteria approach to ranking alternatives. *Journal of the Operational Research Society, 32*, 577–585.

Korhonen, P., Wallenius, J., and Zionts, S. (1984). Solving the discrete multiple criteria problem using convex cones. *Management Science, 30*(11), 1336–1345.

Köksalan, M., and Özpeynirci, S. (2009). An interactive sorting method for additive utility functions. *Computers and Operations Research, 36*(9), 2565–2572.

Koksalan, M., and Tuncer, C. (2009). A DEA-based approach to ranking multi-criteria alternatives. *International Journal of Information Technology & Decision Making, 8*(1), 29–54.

Köksalan, M., and Ulu, C. (2003). An interactive approach for placing alternatives in preference classes. *European Journal of Operational Research*, *144*(2), 429–439.

Köksalan, M., and Sagala, P. N. (1995b). An approach to and computational results on testing the form of a decision maker's utility function. *Journal of Multi-Criteria Decision Analysis, 4* (3), 189–202.

Köksalan, M., and Sagala, P. N. (1995a). Interactive approaches for discrete alternative multiple criteria decision making with monotone utility functions. *Management Science, 41* (7), 1158–1171.

Köksalan, M., Büyükbaşaran, T., Özpeynirci, Ö., and Wallenius, J. (2010). A flexible approach to ranking with an application to MBA Programs. *European Journal of Operational Research, 201*, 470–476.

Köksalan, M., Karwan, M. H., and Zionts, S. (1984). An improved method for solving multiple criteria problems involving discrete alternatives. *IEEE Transactions on Systems, Man, and Cybernetics, 14* (1), 24–34.

Lahdelma, R., Hokkanen, J., and Salminen, P. (1998). SMAA—stochastic multiobjective acceptability analysis. *European Journal of Operational Research*, *106*(1), 137–143.

Manshadi, E. D., Mehregan, M. R., Safari, H. (2015). Supplier Classification Using UTADIS Method Based on Performance Criteria. *International Journal of Academic Research in Business and Social Sciences, 5*(2), 31-45.

Mihelis, G., Grigoroudis, E., Siskos, Y., Politis, Y. and Malandrakis, Y. (2001). Customer satisfaction measurement in the private bank sector. *European Journal of Operational Research, 130*(2), 347-360.

Milton, J.S., and Arnold, J.C., (1995). Introduction to Probability and Statistics. Probability and Statistics, 3rd ed. McGraw-Hill International editions.

Mousseau, V., and Slowinski, R. (1998). Inferring an ELECTRE TRI Model from Assignment Examples, *Journal of Global Optimization, 12*(2), 157-174.

Mousseau, V., Dias, L. C., and Figueira, J. (2003). On the notion of category size in multiple criteria sorting models. *Cahier du LAMSADE 205*, Universite Paris-Dauphine, Paris.

Opricovic, S. (1998). Multi-criteria Optimization of Civil Engineering Systems, *Faculty of Civil Engineering*, Belgrade.

Östermark, R. (1999). A fuzzy neural network algorithm for multigroup classification. *Fuzzy Sets and Systems*, 105, 113-122.

Özpeynirci, Ö., Özpeynirci, S., and Kaya, A. (2017). An interactive approach for multiple criteria selection problem. *Computers & Operations Research*, *78*, 154–162.

Özpeynirci, S., Özpeynirci, Ö., and Mousseau, V. (2018). An interactive algorithm for multiple criteria constrained sorting problem. *Annals of Operations Research*, *267*(1), 447–466.

Pastijin, H., and Leysen, J. (1989). Constructing an Outranking Relation with ORESTE. *Mathematical and Computer Modeling*, *12*(10/11), 1255-1268.

Ramezanian, R. (2019). Estimation of the profiles in posteriori ELECTRE TRI: A mathematical programming model. *Computers & Industrial Engineering*, *128*, 47–59.

Roy, B. (1968). Classement et choix en pr´esence de points de vue multiples (la methode ELECTRE). *RIRO*, *8*:57–75.

Roy, B. (1981). The optimisation problem formulation: criticism and overstepping. *Journal of the Operational Research Society, 32*, 427-436.

Roy, B. (1991). The outranking approach and the foundations of ELECTRE methods. *Theory and Decision*, *31*, 49–73.

Rubinstein, R. (1982). Generating random vectors uniformly distributed inside and on the surface of different regions. *European Journal of Operational Research*, *10*, 205–209.

Saaty, T. L. (1980). *The analytic hierarchy process*. McGraw-Hill, NY.

Savage, L. J. (1951). The theory of statistical decision. *Journal of the American Statistical Association, 46*(253), 55–67.

Shannon, C. (1948). The mathematical theory of communication. *Bell System Technical Journal*, *27*(3), 379–423.

Silberberg, E., and Wing S. (2001). *The Structure of Economics: A Mathematical Analysis*. McGraw-Hill, Irwin.

Sinuany-Stern, Z., Mehrez, A., and Barboy, A. (1994). Academic Departments Efficiency via DEA. *Computers and Operations Research*, *21*(5), 543-556.

Siskos, Y., and Spyridakos, A. (1999). Intelligent multicriteria decision support: Overview and perspectives. *European Journal of Operational Research, 113*, 236-246.

Smith, C. (1947). Some examples of discrimination. *Annals of Eugenics*, 13, 272–282.

Soylu, B. (2011). A multi-criteria sorting procedure with Tchebycheff utility function. *Computers and Operations Research*, *38*(8), 1091–1102.

Spyridakos, A., Siskos, Y., Yannacopoulos, D., and Skouris, A. (2001). Multicriteria Job evaluation for large organizations. *European Journal of Operational Research, 130*, 375–387.

Tervonen, T., and Lahdelma, R. (2007). Implementing stochastic multicriteria acceptability analysis. *European Journal of Operational Research, 178,* 500–513.

Tervonen, T., Figueira, J. R., Lahdelma, R., Dias, J. A., and Salminen P. (2009). A stochastic method for robustness analysis in sorting problems. *European Journal of Operational Research*, *191*(1), 236– 242.

Tervonen, T., van Valkenhoef, G., Baştürk, N., and Postmus, D. (2013). Hit-and-run enables efficient weight generation for simulation-based multiple criteria decision analysis. *European Journal of Operational Research*, *224*(3), 552–559.

Tezcaner Öztürk, D., and Köksalan, M. (2019). An interactive algorithm for multiobjective ranking for underlying linear and quasiconcave value functions. Forthcoming, *International Transactions in Operational Research.*

Torgerson, W. S. (1952). Multidimensional scaling: I. Theory and method. *Psychometrika, 17*, 401-419.

Torgerson, W. S. (1958). *Theory and methods of scaling*. New York: Wiley.

Tryon, R. (1939). *Cluster analysis*. McGraw–Hill, New York.

Tzeng, G. H., and Huang, J. J. (2011). *Multiple attribute decision making: Methods and applications*. Boca Raton, FL, Chapman and Hall/CRC Press, USA.

Ulu, C., and Köksalan, M. (2001). An interactive procedure for selecting acceptable alternatives in the presence of multiple criteria. *Naval Research Logistics*, *48*(7), 592–606.

Ulu, C., and Köksalan, M. (2014). An interactive approach to multicriteria sorting for quasiconcave value functions. *Naval Research Logistics*, *61*, 447–457.

Ulucan, A., and Atıcı, K. B. (2013). A Multiple Criteria Sorting Methodology with Multiple Classification Criteria and An Application to Country Risk Evaluation. *Technological and Economic Development of Economy*, *19*(1), 93-124.

Valkenhoef, G., and Tervonen, T. (2016). Entropy-optimal Weight Constraint Elicitation with Additive Multi-attribute Utility Models. *Omega*, *64*(1), 1–12.

Vetschera, R., Chen, Y., Hipel, K. W., and Kilgour, D. M. (2010). Robustness and information levels in case-based multiple criteria sorting. *European Journal of Operational Research*, *202*(3), 841–852.

Wu, J., Sun, J., Liang, L., and Zha, Y. (2011). Determination of Weights for Ultimate Cross Efficiency Using Shannon Entropy. *Expert Systems with Applications*, *38*(5), 5162–5165.

Xiao, F. (2020). EFMCDM: Evidential Fuzzy Multicriteria Decision Making Based on Belief Entropy. *IEEE Transactions on Fuzzy Systems, 28*(7), 1477–1491.

Yu, W. (1992). ELECTRE TRI: Aspects methodologiques et manuel d'utilisation. *document du LAMSADE, 74,* Université de Paris-Dauphine, Paris.

Zangwill, I. W. (1967) The piecewise concave function. *Management Science, 13*(11), 900–912.

Zhang, H., Gu, C. L., Gu, L. W., and Zhang, Y. (2011). The evaluation of tourism destination competitiveness by TOPSIS & information entropy - A case in the Yangtze River Delta of China, *Tourism Management*, *32*, 443-451.

Zionts, S. (1981). A multiple criteria method for choosing among discrete alternatives. *European Journal of Operational Research, 7*(2), 143–147.

Zopounidis, C., and Doumpos, M. (1999). Business Failure Prediction Using UTADIS Multicriteria Analysis. *Journal of the Operational Research Society*, *50*(11), 1138-1148.

Zopounidis, C., and Doumpos, M. (2000). PREFDIS: A multicriteria decision support system for sorting decision problems. *Computers and Operations Research, 27*(7-8), 779-797.

Zopounidis, C., and Doumpos, M. (2002). Multicriteria classification and sorting methods: A literature review. *European Journal of Operational Research*, *138*(2), 229–246.

Zubin, J. (1938). A technique for measuring likemindedness. *Journal of Abnormal Psychology*, *33*, 508–516.

APPENDICES

A. CURRICULUM VITAE

RESEARCH ASSISTANT ALİ ÖZARSLAN

PhD Candidate, MBA

- Office: I-202, Department of Business Administration, Middle East Technical University, Universiteler Mah., Dumlupinar Blv. No:1, 06800 Çankaya, Ankara, TURKEY
- Phones – Office: +90 (312) 2103033/8
- E-mails: oali@metu.edu.tr / aliozarslan86@gmail.com

PERSONAL INFORMATION

- Birth Date & Place : 29.04.1986 İskenderun / Hatay / TURKEY
- Marital Status : Married, Two Children

EDUCATION

- 2014 - Present Middle East Technical University, Ankara

  Ph.D. in Department of Business Administration (Quantitative Decision Methods Track)
- 2011-2014 Middle East Technical University, Ankara

  M.B.A. with thesis in Department of Business Administration
- 2004-2009 Bilkent University, Ankara

  B.S. in Management in Faculty of Business Administration
- 1997-2004 İstiklal Makzume Anatolian High School, İskenderun/Hatay

PROFESSIONAL EXPERIENCE

 2012-Present Research Assistant at Department of Business Administration, METU

 2011-2012 Research Assistant at Department of Business Administration, Quantitative Methods Track, İstanbul Medeniyet Üniversitesi (ÖYP-Teaching Staff Training Program)

 2009-2011 Research Assistant at Faculty of Business Administration, Bilkent University

AWARDS & ACHIEVEMENTS

 Ranked 44th in the nationwide Academic Staff and Graduate Education Test (2012)

 Full scholarship awarded by Bilkent University for graduate study including tuition waiver, accommodation and monthly stipend. (2009-2011)

 Undergraduate Scholarship given by Bilkent University including tuition waiver, accommodation and monthly stipend. (2004-2009)

 Ranked 1503th (equal weight) in the nationwide University Entrance Examination (ÖSS) out of more than 1,700,000 examinees. (2004)

PUBLICATIONS

Published

 Özarslan, A. and Karakaya, G. (2021). Çok Kriterli Sınıflandırma Problemlerine Yeni Bir Etkileşimli Yöntem: Enerji Sektöründe Bir Uygulama, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 36:4, 2239-2254.

 Özarslan, A. and Gönül, M. S. (2016). Rassal Tavsiyeden Yararlanma ve Karar Performansı, *The Journal of International Management Research*, 2(3), 213-224.

Under Review

 Özarslan, A. and Karakaya, G. (Revised version submitted). Interactive Approaches to Multiple Criteria Sorting Problems: Entropy-Based Question Selection Methods, *International Journal of Information Technology & Decision Making*.

CONFERENCE PRESENTATIONS AND PUBLICATIONS

- Özarslan, A. and Karakaya, G. (2019). An interactive sorting approach based on information theoretic measure, 25th International Conference on MCDM, Istanbul, Turkey.

- Özarslan, A. and Karakaya, G. (2019). Alternatiflerin sınıflandırılması için yeni bir etkileşimli yaklaşım, 39th YAEM National Conference, Ankara, Turkey.

- Esen, Ü. B. and Özarslan, A. (2019). Türkiye'deki Yaratıcı Şehirlerin Belirlenmesi ve Komşu Şehirler Arasındaki Etkileşiminin Mekansal Ekonometri Yöntemi ile İncelenmesi. III. Uluslararası Ekonomi, Finans ve Ekonometri Sempozyumu (ISEFE 2019), Hatay, Turkey.

- Özarslan D. and Özarslan A. (2019). İhracata Başlama ve İhracatı Bırakmanın ARGE Faaliyetleri Üzerine Etkisi. III. Uluslararası Ekonomi, Finans ve Ekonometri Sempozyumu (ISEFE 2019), Hatay, Turkey.

- Özarslan, A. and Özarslan, D. (2018). BIST'te İşlem Gören Şirketlerin Finansal Başarısızlıklarının Tahmini, 38. Yöneylem Araştırması / Endüstri Mühendisliği Ulusal Kongresi (YAEM), Anadolu Üniversitesi Endüstri Mühendisliği Bölümü, Eskişehir, Türkiye.

- Özarslan, A. (2016). Determinants of Non-performing Loans in Central and Eastern European Countries, 2^{nd} International Conference on Applied Economics and Finance, Girne, North Cyprus, 5th-6th December, Conference Abstract proceedings book, 94.

- Özarslan, A. (2016). Exploratory Factor Analysis and Discriminant Analysis on Cleveland Financial Stress Index (CFSI), 2^{nd} International Conference on Social Sciences and Education Research, İstanbul, Turkey, 4^{th}-6^{th} November, The Book of Abstracts, 153.

- Özarslan D. and Özarslan A. (2016). Çok Kriterli Gruplandırma Yöntemleri ile Türkiye’de İflas Eden Firmaların İncelenmesi. International Turgut Özal on Business, Economics and Political Science, Ankara, Turkey.

- Özarslan A. and Özarslan D. (2016). Petrol Fiyatlarıyla BİST Endekslerinin Etkileşimi Üzerine Ampirik bir Çalışma. 1. Lisansüstü İşletme Öğrencileri Sempozyumu, Gaziantep, Turkey.

- Özarslan, A., Gönül M. S. and Nabavi. H. (2015). Çok Amaçlı Karar Verme Yöntemiyle Asfalt Geri Dönüşüm Örneği, YAEM 2015 –35. Yöneylem Araştırması ve Endüstri Mühendisliği Ulusal Kongresi, Ankara, Turkey, 9th-11th September, Özetler: p146.

- Özarslan, A., Nabavi, H. and Gönül, M. S. (2015). Multi-Objective Decision Making in Asphalt Recycling: A Municipality Case, 23rd International Conference on Multiple Criteria Decision Making, Hamburg, Germany, 2nd-7th August, Book of Abstracts: p13.

- Özarslan, A., and Gönül, M. S. (2013). “An empirical study on advice taking: Comparison of techniques on measurement of advice utilization”, 9th International Student Conference, İzmir University of Economics, İzmir, Turkey, April 14-15.

PROJECTS AND INTERNSHIPS

- internship at CMB (Capital Market Board) (February 2009)
- EU Project Management Training Program (November 2007)

COMPUTER SKILLS

- Programming Languages: JAVA, R, E-VIEWS, STATA, SPSS, MINITAB, GPower, SuperDecisions, GLP, MOLP, LINDO, QM for Windows, GAMS, CPLEX, PYTHON, C, C++.

B. TURKISH SUMMARY / TÜRKÇE ÖZET

1. Giriş

Profesyonel veya günlük yaşamda alınan kararlar genellikle birden çok alternatif içermektedir. Bu tür problemler, alternatifler birden fazla kritere göre değerlendirildiğinden çok kriterli karar verme (ÇKKV) problemleri olarak adlandırılır. Alternatifleri değerlendirmek için kullanılan kriterler genellikle birbiriyle çelişmektedir. Örneğin, bir tedarikçi seçim probleminde tedarikçileri değerlendirmek için genelde kalite ve fiyat kriterleri göz önünde bulundurulur. Kaliteli bir ürünün fiyatının genellikle yüksek, kalitesiz bir ürünün fiyatının ise düşük olması beklendiğinden bu iki kriter birbiriyle çelişen kriterler olarak değerlendirilmektedir. Diğer bir örnek ise yatırımcıların yatırım yapacakları finansal araçlara karar verirken getiri ve risk kriterlerini kullanmasıdır.

Birbiriyle çelişen kriterler ile değerlendirilen alternatiflerin olduğu problemlerde karar vericinin (KV'nin) bu kriterler arasında ödünleşim yapması gerekmektedir. Bu tür yargısal kararlar KV'nin tercih yapısı hakkında bilgi sağlamaktadır. Çok kriterli karar yardımı (ÇKKY) yöntemleri karar verme sürecine yardımcı olmak için KV'nin tercih bilgilerini kullanır. Bu yöntemlerdeki amaç, örnek kararlar yoluyla KV'nin tercih yapısını ortaya çıkararak çözüme ulaşmaktır. Roy (1981), ÇKKV problemlerini tanımlama, seçim, sıralama ve sınıflandırma problemleri olarak dörde ayırmıştır. Tanımlama problemleri, problemin özelliklerini tanımlamak için alternatiflerin ve benzersiz özelliklerinin açığa çıkarıldığı problemlerdir. Seçim problemlerinde amaç, en çok tercih edilen alternatifi veya alternatifler kümesini seçmektir. Seçim problemlerine tipik bir örnek, bir şirketteki bir pozisyon için en uygun aday(lar)ı belirlemektir. Diğer bir örnek ise en uygun tedarikçi veya tedarikçileri seçmektir. Sıralama problemlerinde ise alternatifler en çok tercih edilenden en az tercih edilene doğru sıralanmaktadır. Enerji sürdürülebilirliğine dayalı ülke sıralamaları ve üniversite sıralamaları bu problemlerin tipik örnekleridir.

Sınıflandırma problemlerinde ise KV'nin alternatifleri iki veya daha fazla kategoriye ayırması gerekmektedir. Burada amaç, daha iyi kategorideki bir alternatif, daha kötü kategorilerdeki tüm alternatiflere tercih edilecek şekilde alternatifleri kategorilere atamaktır. Kategoriler, eşik değerler adı verilen sınırlarla birbirinden ayrılmaktadır. Sınıflandırma problemlerine örnek olarak kredi başvurularının kabul edilmiş, beklemede ve reddedilmiş olarak üç kategoriye ayrılması verilebilir. Kredi derecelendirme kuruluşları tarafından firmaların risk tutumlarına göre kategorilere ayrılması bir başka örnektir.

Çok kriterli sınıflandırma (ÇKS) problemlerini çözmek için geliştirilen farklı yaklaşımlar ve yöntemler bulunmaktadır. Bu yaklaşımlardan biri olan tercih ayrıştırma analizi KV'nin tercihleriyle tutarlı bir tercih modeli oluşturmaktadır (Jacquet-Lagreze ve Siskos, 2001). Oluşturulacak tercih modelinin parametrelerini bulmak için KV'den alınan karar örnekleri kullanılmaktadır. KV'nin tercihlerini kullanmanın bir yolu, alternatifleri atamak için gerekli model parametrelerinin değerlerinin KV tarafından belirlemesidir. Buna doğrudan ortaya çıkarma tekniği denmektedir ve genellikle bu durumda KV'nin bilişsel yükü oldukça fazladır. ELECTRE-TRI yöntemi ÇKS problemleri için yaygın olarak kullanılan bir doğrudan ortaya çıkarma tekniğidir (Yu, 1992). Yöntem, KV'nin karar ağırlıklarının, kategori eşiklerinin ve temsili kategori profillerinin belirlenmesini gerektirmektedir. Böylece alternatifler arasında ikili karşılaştırmalar yapılarak sınıflandırma yapılmaktadır.

Doğrudan ortaya çıkarma tekniklerine bir alternatif olarak KV'nin tercih yapısının tercih fonksiyonu olarak adlandırılan bir yapıya sahip olduğu varsayılmaktadır. Bu tür yaklaşımlara fonksiyon tabanlı yaklaşımlar denmektedir. Fonksiyon tabanlı yaklaşımlar, KV'den elde edilen karar örneklerine dayalı olarak model parametrelerini bulmaya çalışmaktadır. Bunun bir yolu KV'nin başlangıçta atama örneklerini sağlaması (örn., 3. alternatif birinci kategoriye atansın, 5. alternatif ikinci kategoriye atansın gibi) ve bu örnek atamaları kullanarak oluşturulan karar modeli tarafından geri kalan alternatiflerin atanmasıdır. Atama örnekleri elde etmenin başka bir yolu is KV ile interaktif bir şekilde tercih yapısının aşamalı olarak ortaya çıkarılmasıdır. İnteraktif yaklaşımlarda KV'nin karar sürecine dâhil olması, model parametrelerinin KV tarafından aşamalı olarak öğrenilmesini sağlamaktadır. Literatürde, KV'nin tercihlerinin farklı toplamsal tercih fonksiyonları ile tutarlı olduğu varsayılarak çeşitli sınıflandırma yaklaşımları geliştirilmiştir. Toplamsal tercih fonksiyonlarında her bir kriterdeki fayda değerlerinin toplanması ile alternatiflerin nihai fayda değerleri belirlenmektedir. Genel monoton (Greco vd., 2011), parçalı doğrusal (Köksalan ve Özpeynirci, 2009) ve yarı-içbükey (Ulu ve Köksalan, 2001; 2014) tercih fonksiyonları KV'nin tercihlerini temsil etmek için yaygın olarak kullanılan toplamsal fonksiyon biçimleridir.

Bu tezde, KV'nin tercihlerinin parçalı doğrusal ve genel monoton toplamsal tercih fonksiyonları ile tutarlı olduğunu varsayılarak KV ile interaktif sınıflandırma yaklaşımları geliştirilmektedir. Alternatiflerin atanabilecekleri kategorileri belirlemek için matematiksel modeller çözülmektedir. Her iterasyonda KV'den bir alternatifin kategori bilgisi alınmaktadır. KV'den alınan atama bilgisi matematiksel modellere dâhil edilerek diğer alternatiflerin kategori aralığını daraltmada kullanılmaktadır. Matematiksel modellerden ve Monte Carlo simülasyonlarından elde edilen ve KV'nin tercihleriyle uyumlu olan parametreleri kullanarak alternatifler kategorilere farazi olarak atanmaktadır. Burada amaç alternatiflerin kategorilere atanma sıklıkları hakkında bilgi toplamaktır. Kategorilere atanma sıklıkları alternatiflerin bir kategoriye ait olma olasılığına dönüştürülmektedir. KV'ye sorulacak alternatifi seçmek ve sistemin belirsizliğini belirlemek için bir bilgi teorik ölçüsü olan göreceli entropi kullanılmaktadır. Önerilen olasılıksız ve olasılıksal algoritmaların performansını ölçmek için literatürden üç örnek problem ve rassal veri ile oluşturulmuş problemler üzerinde farklı yöntemlerle kıyaslama yapılmaktadır.

Tezin geri kalanı şu şekilde organize edilmiştir: Bölüm 2'de, ÇKKY yaklaşımlarına ilişkin literatür taraması verilmektedir. Bölüm 3'te ilk olarak, geliştirilen etkileşimli yaklaşımlar sunulmaktadır. Daha sonra kıyaslama amacıyla kullanılan yaklaşımlar açıklanmaktadır. Bölüm 4'te ise geliştirilen ve kıyaslama amacıyla kullanılan yaklaşımların performanslarını ölçmek için yapılan uygulamaların sonuçları ile kısıtlamalar ve gelecekteki çalışma alanları anlatılmaktadır.

2. Literatür Taraması

Sınıflandırma problemleri, seçim ve sıralama problemlerinden farklı özelliklere sahiptir (Vetschera vd., 2010). Seçim ve sıralama problemlerinde yeni alternatiflerin eklenmesi, mevcut alternatiflerin mevcut konumunu değiştirebilir. Sınıflandırma problemlerinde ise yeni alternatiflerin eklenmesi, önceden atanan alternatiflerin kategorisini değiştirmez (Zopounidis ve Doumpos, 2002). Bu nedenle sınıflandırma problemleri genellikle KV'nin mutlak kararlar vermesini gerektirirken, kararlar seçim ve sıralama problemlerinde görecelidir. Ayrıca, sınıflandırma problemlerinde seçim ve sıralama problemlerinde kullanılan alternatifler arasında ikili karşılaştırmalar yerine genellikle atama örnekleri kullanılır.

Doumpos ve Zopounidis (2011) yaygın olarak kullanılan ÇKKY yaklaşımlarını istatistiksel öğrenme ve tercih ayrıştırma analizi olarak iki kısma ayırmaktadır. İstatistiksel öğrenme tekniklerin kural tabanlı modeller sıralama problemleri için sıklıkla kullanılmaktadır (Greco vd., 2001; 2002). Tercih ayrıştırma analizi ise KV'nin tercihlerine dayalı olarak kriter değerlerinin birleştirilmesi üzerine tercih modelinin oluşturulmasını içermektedir (Jacquet-Lagreze ve Siskos, 2001). Finansal yönetim (Zopounidis vd., 2000), pazarlama (Mihelis vd., 2001) ve iş değerlendirmesi (Spyridakos vd., 2001) gibi alanlarda tercih ayrıştırma analizi ile ilgili uygulamalar yapılmaktadır.

ÇKKV problemlerine fonksiyon tabanlı yaklaşımlarda KV'nin tercihlerinin bir tercih fonksiyonu ile tutarlı olduğu varsayılmaktadır. Toplamsal tercih fonksiyonu tercih modellemesi için yaygın olarak kullanılmaktadır (Keeney ve Raiffa, 1976). Köksalan ve Sagala (1995b), KV'nin tercihleriyle tutarlı olan tercih fonksiyonunun formunu test etmek için interaktif bir yaklaşım geliştirmiştir. Doğrusal, yarı-içbükey, yarı-dışbükey veya genel monoton toplamsal tercih fonksiyonu ile ilgili olarak KV'nin tercihlerinin tutarlılığı aşamalı olarak aranmaktadır. Her iterasyonda KV'den alternatifler arasında ikili karşılaştırmalar yapması istenmektedir ve bu bilgi fonksiyonların tutarlılığını kontrol etmek için kullanılmaktadır.

Doğrusal tercih fonksiyonu, KV'nin tercihlerini temsil etmek için pratik ve kullanışlı bir fonksiyondur. Kriterlerin ağırlıkları karşılık gelen kriter puanları ile çarpılıp toplanarak alternatiflerin toplam faydası bulunmaktadır. Doğrusal tercih fonksiyonuna ilişkin en yaygın ve temel yaklaşımlardan biri her bir kriterde sabit ağırlıkların kullanılmasıdır. Ağırlıklar genellikle yetkililer veya uzmanlar tarafından belirlenir. Örneğin, global MBA programı sıralamaları Financial Times (FT) tarafından yıllık olarak yayınlanmaktadır ve mezun maaşı, kadın öğretim üyelerinin oranı gibi 20 kriter bulunmaktadır. Köksalan vd. (2010) sabit ağırlıklar yerine ağırlık aralıklarının kullanılmasını önermektedir. Sabit ağırlıklardaki küçük değişikliklerin kararlarda dramatik değişikliklere neden olduğu sonucuna ulaşılmıştır.

Yarı-içbükey tercih fonksiyonları ise çoğu gerçek dünya durumunda insan davranışlarını iyi temsil etmektedir (Arrow ve Enthoven, 1961). Yatırımlarda riskten kaçınma ve tüketimde azalan marjinal ikame oranı gibi prensipler, yarı-içbükeylik varsayımıyla açıklanmaktadır (Silberberg ve Suen, 2001, s. 260-261). Yarı-içbükey fonksiyonlar, parçalı bir biçimde doğrusal bir fonksiyonla yaklaşık olarak tahmin edilebilir (Zangwill, 1967). İyi bilinen bir sınıflandırma yöntemi olan UTADIS, Devaud vd. (1980) ve Jacquet-Lagreze (1995) tarafından geliştirilmiştir. UTADIS'te alternatifler parçalı doğrusal tercih fonksiyonu tahmin edilerek kategorilere atanmaktadır. Alternatiflerin yanlış sınıflandırılmasından kaynaklanan sınıflandırma hatalarını en aza indirmek için bir doğrusal programlama (DP) modeli çözülmektedir. UTADIS kategorilerin eşik değerleriyle ayrıldığı eşik tabanlı bir sınıflandırma yöntemidir. KV'den alınan referans alternatiflerin atama bilgileri DP modeline eklenerek kriter ağırlıkları ve kategori eşik değerleri bulunmaktadır. Bu değerler kullanılarak alternatifler kategorilere atanmaktadır. UTADIS, KV'den elde edilen tercih bilgilerine iyi uyması amaçlanan tek bir parametre kümesini tahmin etmektedir. Ancak, alternatiflerin farklı sınıflandırılmasına neden olan KV'nin tercihleriyle uyumlu birden çok parametre seti olabilir (Köksalan ve Özpeynirci, 2009). Tek bir tercih fonksiyonundan türetilen karar parametrelerinin sağlamlığı, sıralı regresyon çerçevesinde dikkate alınmıştır (Figueira vd., 2009; Greco vd., 2008; 2010). Fonksiyon tabanlı yaklaşımlarda sıralı regresyonun rolü, KV'nin tercihlerine dayalı olarak sağlam sonuçlar formüle etmektir. Sıralı regresyon tekniği, UTA ve UTADIS yöntemlerinde olduğu gibi KV'nin karar örnekleriyle uyumlu tek bir tercih fonksiyonu yerine uyumlu tercih fonksiyonlarının bütününü dikkate almaktadır.

İnteraktif yaklaşımlar genellikle KV'nin tercih fonksiyonunun örtük biçimde olduğunu varsaymaktadır (Korhonen vd., 1992). Dolayısıyla bu yaklaşımlar, uyumlu tercih fonksiyonlarının tamamını dikkate almaktadır (Greco vd., 2010). KV ile etkileşimli olarak tercih bilgilerinin aşamalı bir şekilde ortaya çıkarılması kişinin tercih yapısı hakkında öğrenmeyi geliştirmeye yardımcı olmaktadır. Özellikle atamalarda tutarsızlıklar olması durumunda kararların yeniden gözden geçirilmesi için KV'nin önceki adımlara dönmesi sağlanmaktadır. ÇKS problemlerine uyumlu tercih fonksiyonlarının tamamını dikkate alan öncü yaklaşımlardan biri Ulu ve Köksalan'ın (2001) çalışmasıdır. Çalışma, KV'nin tercihlerinin doğrusal, yarı-içbükey veya genel monoton tercih fonksiyonu ile tutarlı olduğunu varsayarak alternatifleri iki kategoriye atayan etkileşimli yaklaşımlar geliştirmektedir. Çalışma, çözülen DP modellerinin sayısını azaltmak için alternatiflerin kategorileri aralıklarını tanımlarken baskınlık ilişkilerinden yararlanmaktadır.

Köksalan ve Ulu (2003), Ulu ve Köksalan'ın (2001) yaklaşımını doğrusal tercih fonksiyonu varsayımıyla ikiden fazla kategori için genelleştirmektedir. Alternatiflerin olası kategorileri atanabilecekleri en iyi ve en kötü kategorileri açısından tanımlanmaktadır. Alternatiflerin ait olduğu gerçek kategorileri bulmak için DP modelleri ile alternatiflerin kategori aralığı daraltılırken her iterasyonda KV'den bir alternatifin atama bilgisi istenmektedir. Köksalan ve Özpeynirci (2009) ise KV'nin parçalı doğrusal bir fayda fonksiyonuna sahip olduğunu varsayarak karma tamsayılı programlama (KTP) modellerini kullanarak eşik tabanlı etkileşimli bir sıralama yaklaşımı geliştirmiştir. Alternatiflerin olası atamalarını belirlemek için matematiksel programlama modellerinin kullanımı sağlam sıralı regresyon (SSR) ilkesinde genelleştirilmiştir (Greco vd., 2008). SSR ilkesi, KV'nin tercihleriyle uyumlu tüm olası parametre kümelerini dikkate almaktadır.

Benabbou vd. (2017), eşik tabanlı sınıflandırma problemleri için Choquet integraline dayalı KV ile interaktif olarak pişmanlık tabanlı bir yaklaşım geliştirmiştir. Choquet integralleri, etkileşimli kriterlere ve toplamsal olmayan tercihlere izin vermektedir (Choquet, 1955; Denneberg, 1994). Çalışma kategorileri ayıran eşiklerin karar sürecinin başında KV tarafından tanımlandığını varsaymaktadır. Pişmanlık yaklaşımı, eşikler ve Choquet değerleri arasındaki farka dayalı olarak alternatiflerin yanlış atanmasını dikkate almaktadır. Yanlış sınıflandırmadan kaynaklanan pişmanlığı en aza indiren atamayı bulmak için DP modelleri çözülmektedir. Minimaks pişmanlık stratejisinin KV'den bilgi toplama sürecinde kullanılması önerilmektedir.

Bazı sınıflandırma problemlerinde KV'nin belirgin tercihleri veya problemin doğası gereği bazı sınırlamalar hakkında karar sürecinin başında bilgi sahibi olunabilmektedir. Örneğin, insan kaynakları yöneticisi işe başvuranları işe alınan ve alınmayan olarak sınıflandırırken deneyime dayalı kriterlerin eğitime dayalı kriterlerden daha önemli olduğunu belirtebilir. Ayrıca problemin doğası gereği kategori büyüklüğü kısıtlamaları olarak bir kategoriye/kategorilere atanacak alternatiflerin sayısı üzerinde belirli sınırlar veya kesin değerler getirebilir. Örneğin, bir kredi yöneticisi, kabul edilecek kredi başvurularının sayısına kısıtlama getirebilir. Bu tür problemler, Mousseau vd. (2013) tarafından kısıtlı sıralama problemleri (KSP) olarak tanımlanmaktadır. Kriterlerin önemi üzerindeki kısıtlamalar, fonksiyon tabanlı yaklaşımlarda matematiksel modellere doğrusal kısıtlamalar eklenerek ele alınabilir. Fakat kategori büyüklüğü kısıtlamaları bir kategoriye atanacak alternatif sayısını sınırlamak için ikili değişkenlerin tanımlanmasına ihtiyaç duyar.

Sınıflandırma problemleri üzerine yapılan güncel çalışmalarda alternatiflerin kategorilere atanma olasılığı üzerine yoğunlaşılmaktadır. Kadzinski ve Tervonen (2013), Greco vd. (2008; 2010) çalışmalarındaki SSR yaklaşımı ile stokastik çok kriterli kabul edilebilirlik analizini (SÇKKA) birleştirmişlerdir. Lahdelma vd. (1998) tarafından geliştirilen SÇKKA, KV'nin tercihlerini temsil eden karar ağırlık uzayı hakkında bilgi sağlayan bir simülasyon tekniğidir. KV'nin tercihleri ile uyumlu tercih fonksiyonlarının örneklemesi Monte Carlo simülasyonları ile elde edilmektedir. Çalışmada kategoriye ait olma olasılığını ifade eden kategori kabul edilebilirlik indeksi (KKİ) geliştirilmiştir. Atama örneklerinin sayısı arttıkça örnekleme için oluşturulan tercih fonksiyonlarının uyumsuzluk sebebi ile çoğunlukla kullanılamadığı/reddedildiği sonucuna ulaşılmaktadır.

Buğdacı vd. (2013) KV'nin tercihlerinin parçalı doğrusal toplamsal fonksiyonla tutarlı olduğunu varsayarak alternatifleri sıralamak için etkileşimli bir olasılıksal yaklaşım önermektedir. Alternatiflerin kategori aralıklarını daraltmak için DP modelleri çözülmektedir. Kriter ağırlıkları ve kategori eşikleri bilinmeyen parametreler olarak kabul edilerek bunların minimum ve maksimum değerleri ek DP'ler aracılığıyla tahmin edilmektedir. Minimum ve maksimum değerler arasında tekdüze dağılım olduğu varsayımıyla alternatiflerin fayda değerinin kategori eşiğinden daha büyük olma olasılığı hesaplanmaktadır. Olasılık değeri KV tarafından belirlenen bir eşik değer olan 1-τ değerini aşıyorsa algoritma alternatifi ilgili kategoriye olasılıksal olarak atamaktadır. Her iterasyonun sonunda KV tarafından atanmak üzere bir alternatif seçilmektedir. Olasılıksal sınıflandırmaya ek olarak kategorilere alternatifleri doğru bir şekilde yerleştiren olasılıksız durumu da dikkate almaktadır. KV'ye sorulacak alternatifin seçimi de olasılıklara dayanmaktadır. Olasılığı 0,5'e en yakın olan alternatif, KV'ye sorulacak en belirsiz alternatif olarak kabul edilmektedir.

3. Sınıflandırma Problemlerine İnteraktif Yaklaşımlar

Bu tezde çoklu kriterlere göre değerlendirilen alternatifleri sınıflandırmak için KV ile etkileşimli yaklaşımlar geliştirilmektedir. KV'nin tercihlerinin (i) parçalı doğrusal ve (ii) genel monoton toplamsal tercih fonksiyonları ile temsil edildiğini varsayarak matematiksel modeller çözülmektedir. Kategori büyüklüğü kısıtlamalarının olduğu durumda ise DP modelleri, KTP modellerine dönüşmektedir. Olasılıksız ve olasılıksal durumlar için farklı algoritmalar oluşturulmaktadır. Ayrıca algoritmaların sınıflandırma performansını ölçmek için kullanılan kıyaslama algoritmaları da açıklanmaktadır.

3.1 Parçalı Doğrusal Tercih Fonksiyonları

Bu bölümde KV'nin tercihlerinin parçalı doğrusal bir fayda fonksiyonu ile tutarlı olduğu varsayılmaktadır. $DP1_{a_t,k}$ ve $DP2_{a_t,k}$ kategori aralıklarını belirlemek için kullanılan enküçükleme ve enbüyükleme modelleridir. A kümesinin n kritere göre değerlendirilen m alternatifi $a_1, a_2, \ldots, a_m$ şeklinde gösterilmektedir. KV'nin alternatifleri $C_1, C_2, \ldots, C_q$ diye gösterilen q adet kategoriye ataması gerekmektedir. Burada C_1 ve C_q, en çok ve en az tercih edilen kategorilerdir. C_k, k kategorisine ait alternatifler kümesini temsil ederken C_0 kategorileri bilinmeyen alternatifler kümesidir.

$(DP1_{a_t,k})$

$$Min\, U(a_t) - u_k \tag{3.1}$$

$Kısıtlar$:

$$U(a_j) = \sum_{i=1}^{n} \left(\sum_{p=1}^{r_{ji}-1} w_{ip} + \frac{g_i(a_j) - g_i^{r_{ji}}}{g_i^{r_{ji}+1} - g_i^{r_{ji}}} w_{ir_{ji}} \right), \forall a_j \in A \tag{3.2}$$

$$U(a_j) \geq u_k,\ \forall a_j \in C_k, \quad k = 1, \ldots, q-1 \tag{3.3}$$

$$U(a_j) \leq u_{k-1} - \delta,\ \forall a_j \in C_k, \quad k = 2 \ldots, q \tag{3.4}$$

$$u_{k-1} - u_k \geq \delta, \quad k = 2, \ldots, q-1 \tag{3.5}$$

$$u_{q-1} \geq \delta \tag{3.6}$$

$$u_1 \leq 1 - \delta \tag{3.7}$$

$$\sum_{i=1}^{n} \sum_{p=1}^{b_i-1} w_{ip} = 1 \tag{3.8}$$

$$w_{ip} \geq 0, \quad i = 1, \ldots, n, \quad p = 1, \ldots, b_i \tag{3.9}$$

Burada δ, 0,001 olarak ayarlanmış küçük bir pozitif sabittir. Modelin amaç fonksiyonu, alternatif a_t'nin faydası ile k kategorisinin eşik değeri arasındaki farkı en aza indirmektir. Kısıt (3.2), parçalı bir doğrusal toplamsal fonksiyona dayalı olarak her bir alternatifin faydasını belirlemektedir. (3.3) ve (3.4), KV tarafından atanan alternatiflerin kategori sınırları içinde olmasını sağlamaktadır. Kısıt (3.5), daha iyi bir kategorinin fayda eşiğinin daha kötü bir kategoriden daha yüksek olduğunu garanti etmektedir. (3.6) ve (3.7) kısıtlarında kategori eşiklerinin alt ve üst limitleri, en çok ve en az tercih edilen kategorilerin fayda aralıklarına sahip olacak şekilde belirlenir. Kısıt (3.8), w_{ip} değerlerinin toplamının bire eşit olmasını sağlamaktadır ve kısıt (3.9) negatif olmama kısıtıdır. İkinci modelde, (3.2)'den (3.9)'a kadar olan tüm kısıtlar aynı kalarak amaç fonksiyonunu enbüyükleme şeklinde değiştirilmektedir.

$\left(DP2_{a_t,k}\right)$

$$Maks\ U(a_t) - u_k \tag{3.10}$$

$s.t.\quad (3.2) - (3.9)$

Kategori büyüklüğü kısıtlamalarını modellere yansıtmak için kategorilerdeki alternatif sayıları matematiksel modellere yansıtılmıştır. Bu tür kısıtlamaları dahil edebilmek için kullanılan y_{jk} ikili değişkeni, alternatif a_j k kategorisine atanmışsa bir değerini, aksi takdirde sıfır değerini alacak şekilde tanımlanmıştır. Oluşturulan KTP modelleri aşağıda verilmektedir.

$\left(KTP1_{a_t,k}\right)$

$Min\ U(a_t) - u_k$

$s.t.\quad (3.2) - (3.9),$

$$U(a_j) \geq u_k - M(1 - y_{jk}),\ \forall a_j \in C_0,\quad k = 1, \dots, q-1 \tag{3.11}$$

$$U(a_j) \leq u_{k-1} + M(1 - y_{jk}) - \varepsilon,\ \forall a_j \in C_0,\quad k = 2 \dots, q \tag{3.12}$$

$$\sum_{k=1}^{q} y_{jk} = 1 \quad \forall a_j \in A \tag{3.13}$$

$$\sum_{j=1}^{m} y_{jk} = s_k \quad \forall C_k \in C^S \tag{3.14}$$

$$y_{jk} \in \{0, 1\},\qquad \forall j, k \tag{3.15}$$

Burada M büyük bir pozitif sabittir. s_k ise k kategorisinde bulunan alternatif sayısıdır ve C^S kesin kategori büyüklüğü bilinen kategoriler kümesidir. (3.11) ve (3.12) kısıtları ikili değişkenler aracılığıyla kategorisi henüz bilinmeyen alternatiflerin atandıkları kategori sınırları içinde fayda değeri almalarını sağlamaktadır. Kısıt (3.13) her alternatifin tek bir kategoriye atanabileceğini belirtmektedir. Kategori büyüklüğü kısıtlamaları, kısıt (3.14) içinde ele alınmaktadır. İkinci modelde, (3.2)'den (3.9)'a ve (3.11)'den (3.15)'e kadar tüm kısıtlamalar korunarak enbüyükleme modeli çözülmektedir.

$(KTP2_{a_t,k})$

$Maks\ U(a_t) - u_k$

$s.t. \quad (3.2) - (3.9), (3.11) - (3.15)$

3.2 Genel Monoton Tercih Fonksiyonları

KV'nin tercihlerinin monoton azalmayan biçimde toplamsal fayda fonksiyonu ile tutarlı olduğu varsayıldığında kriter skorları en az tercih edilenden en çok tercih edilene doğru sıralanmaktadır. $x_1^i, x_2^i, \ldots, x_{m_i}^i$ kriter g_i'deki skorların sıralanmış halini ifade etmektedir ($x_h^i < x_{h+1}^i$, $h=1,\ldots,m_i-1$, $m_i \leq m$). g_i kriterindeki alternatif a_j'nin marjinal faydası $u_i\left(g_i(a_j)\right)$ olarak tanımlansın. Burada alternatif a_j'nin nihai faydası, $U(a_j)$, her bir kriterdeki marjinal faydaların toplamıdır.

$(DP3_{a_t,k})$

$Min\ U(a_t) - u_k$

$s.t. \quad (3.3) - (3.7),$

$$u_i(x_h^i) \leq u_i(x_{h+1}^i), h = 1, \ldots, m_i - 1, \qquad i = 1, \ldots, n \tag{3.16}$$

$$u_i(x_1^i) = 0, \ \ i = 1, \ldots, n \tag{3.17}$$

$$\sum_{i=1}^{n} u_i(x_{m_i}^i) = 1 \tag{3.18}$$

$$U(a_j) = \sum_{i=1}^{n} u_i\left(g_i(a_j)\right), \ \forall a_j \in A \tag{3.19}$$

Kısıt (3.16) daha küçük kriter skorlarının her bir kriterde daha düşük marjinal faydalara sahip olmasını garanti etmektedir. Kısıtlar (3.17) ve (3.18) marjinal fayda değerlerinin [0,1] aralığında olmasını sağlamaktadır. Kısıt (3.19)'da alternatiflerin nihai fayda değerleri hesaplanmaktadır. Aşağıda enbüyükleme amaç fonksiyonuna sahip DP modeli ve KTP modelleri yer almaktadır.

$\left(DP4_{a_t,k}\right)$
$Maks\ U(a_t) - u_k$
$s.t.$ $\quad$ (3.3) − (3.7), (3.16) to (3.19)

$\left(KTP3_{a_t,k}\right)$
$Min\ U(a_t) - u_k$
$s.t.$ $\quad$ (3.3) − (3.7), (3.11) − (3.19)

$\left(KTP4_{a_t,k}\right)$
$Maks\ U(a_t) - u_k$
$s.t.$ $\quad$ (3.3) − (3.7), (3.11) − (3.19)

3.3 Olasılıksız Durum için Algoritma

a_t alternatifinin en kötü ve en iyi kategorileri C_t^{EK} ve $C_t^{Eİ}$ ile gösterilmektedir. $DP1_{a_t,k}$ ve $DP2_{a_t,k}$ modellerinin optimal hedef fonksiyonu değerleri ise $hf_1^*(a_t,k)$ ve $hf_2^*(a_t,k)$ ile temsil edilmektedir. Başlangıçta KV'den atama bilgisi alınmadığı için $C_0 = A$ ve $C_k = \emptyset,\ k = 1,\dots,q$ olarak algoritma başlamaktadır. Her $a_t \in C_0$ için başlangıçta $C_t^{\mathrm{EK}} = q$ ve $C_t^{\mathrm{Eİ}} = 1$ olarak tanımlanmaktadır. Aşağıda olasılıksal atamalara izin verilmeyen durum için algoritmanın (A_{RENT}) aşamaları verilmektedir. Birinci aşamada her alternatif için matematiksel modeller çözülerek kategori aralıkları belirlenmektedir. Burada süreci kısaltmak adına en kötü kategoriyi tespit ederken alternatiflerin en iyi kategorisinden başlanmaktadır. Bir alternatifin kategorisi daraltıldığı zaman baskınlık ilişkileri kullanılarak diğer alternatifler için de kategori daraltma işlemi yapılmaktadır. İkinci aşamada ise KV'ye kategorisi sorulacak bir alternatif seçilmektedir. KV'ye sorulmak üzere seçilen alternatifin kategorisini belirledikten sonra baskınlık ilişkilerini kullanarak diğer alternatiflerin kategori aralıkları daraltılmaktadır. KV'den alınan bilgi DP modellerine eklenerek yeniden birinci aşamaya geçilmektedir. Tüm alternatifler kategorilere yerleştirilene kadar aynı aşamalar uygulanmaktadır. Son aşamada tüm atama bilgileri KV'ye sunularak algoritma sonlanmaktadır.

Algoritma A_{RENT}

Aşama 1 (*kategori daraltma*): Her $a_t \in C_0$ için $k = C_t^{\text{Eİ}}$ olsun.

1.1. $DP1_{a_t,k}$ modelini çöz.

- Eğer $hf_1^*(a_t, k) \geq 0$ ise $C_t^{\text{EK}} = k$ olarak güncelle. a_t'ye baskın olan her $a_{t'} \in C_0$ için $C_{t'}^{\text{EK}} = k$ olarak güncelle. $C_{t'}^{\text{EK}} = C_{t'}^{E\text{İ}} = k$ ise $C_k \leftarrow C_k \cup \{a_{t'}\}$ ve $C_0 \leftarrow C_0 - \{a_{t'}\}$ olarak güncelle. $C_t^{\text{EK}} = C_t^{\text{Eİ}} = k$ ise $C_k \leftarrow C_k \cup \{a_t\}$ ve $C_0 \leftarrow C_0 - \{a_t\}$ olarak güncelle ve aşama 1.3'e git. Aksi halde k = k – 1 olsun ve aşama 1.2'ye git.
- Eğer $hf_1^*(a_t, k) < 0$ ve $k < C_t^{\text{EK}} - 1$ ise k = k + 1 olsun ve aşama 1.1'i tekrar et.
- Eğer $hf_1^*(a_t, k) < 0$ ve $k = C_t^{\text{EK}} - 1$ ise aşama 1.2'ye git.

1.2. $DP2_{a_t,k}$ modelini çöz.

- Eğer $hf_2^*(a_t, k) < 0$ ise $C_t^{\text{Eİ}} = k + 1$ olarak güncelle. a_t'nin baskın olduğu her $a_{t'} \in C_0$ için $C_{t'}^{E\text{İ}} = k + 1$ olarak güncelle. $C_{t'}^{\text{EK}} = C_{t'}^{E\text{İ}} = k + 1$ ise $C_k \leftarrow C_k \cup \{a_{t'}\}$ ve $C_0 \leftarrow C_0 - \{a_{t'}\}$ olarak güncelle. $C_t^{\text{EK}} = C_t^{\text{Eİ}} = k + 1$ ise $C_{k+1} \leftarrow C_{k+1} \cup \{a_t\}$ ve $C_0 \leftarrow C_0 - \{a_t\}$ olarak güncelle. Aşama 1.3'e git.
- Eğer $hf_2^*(a_t, k) \geq 0$ ve $k > C_t^{\text{Eİ}}$ ise k = k – 1 olarak güncele ve aşama 1.2'yi tekrar et.
- Eğer $hf_2^*(a_t, k) \geq 0$ ve $k = C_t^{\text{Eİ}}$ ise aşama 1.3'e git.

1.3. Eğer C_0'daki tüm alternatifler için kategori daraltma aşaması uygulandıysa, aşama 2'ye git. Aksi halde sıradaki $a_t \in C_0$ ile aşama 1.1'e git.

Aşama 2 (*KV'ye sorulacak alternatifi belirleme*): KV'ye sormak için $a_t^s \in C_0$ alternatifini seç. a_t^s'nin KV tarafından atandığı kategori $C_{k'}$ olsun. $C_{k'} \leftarrow C_{k'} \cup \{a_t^s\}$ ve $C_0 \leftarrow C_0 - \{a_t^s\}$ olarak güncelle. a_t^s'ye baskın olan her $a_{t'} \in C_0$ için $C_{t'}^{EK} = k'$ olarak; a_t^s'nin baskın olduğu her $a_{t'} \in C_0$ için $C_{t'}^{\text{Eİ}} = k'$ olarak güncelle. Eğer $C_0 = \emptyset$ ise aşama 3'e git. Aksi halde aşama 1'e git.

Aşama 3 (*bitiş*): Tüm alternatiflerin atama bilgilerini KV'ye bildir ve dur.

Bu çalışmada alternatiflerin kategorilere atanma eğilimini belirlemek için model tabanlı ve simülasyon tabanlı olmak üzere iki farklı yaklaşım geliştirilmektedir. Model tabanlı yaklaşımda modellerin parametre değerleri kullanılmaktadır. Kategorisi kesin olarak bilinmeyen m tane alternatifin q tane kategoriye atanması probleminde alternatiflerin kategorilere atanma durumunu belirlemek için $DP1_{a_t,k}$ ve $DP2_{a_t,k}$ modelleri çözülmektedir. Bu da toplamda 2*m*(q-1) DP modeli çözüldüğünü göstermektedir. Bir alternatif için $DP1_{a_t,k}$ modeli çözüldüğünde u_k sınır değerinin en büyük değeri elde edilirken $DP2_{a_t,k}$ modeli çözüldüğünde en küçük değeri elde edilmektedir. KV'den bilgi alındıkça en büyük ve en küçük değerler arasındaki farkın azalması ve sınır değerlerin gerçek değerlerine yakınsamaları beklenmektedir. Kategorilerin sınır değerlerinin en büyük ve en küçük değerleri arasında tekdüze dağılım gösterdiği varsayılarak bu iki değerin ortalaması alınmaktadır. Kategorisi bilinmeyen alternatifler için her modelde farklı w_{ip} ve $U(a_j)$ değerleri bulunmaktadır. $DP1_{a_t,k}$ ve $DP2_{a_t,k}$ modelleri çözüldüğünde sınır değerlerinin ortalaması alınırken alternatiflerin de toplam fayda değerlerinin ortalaması alınmaktadır. İki model çözüldüğünde ortalama sınır değerlerine ve alternatiflerin ortalama toplam fayda değerlerine göre farazi sınıflandırma yapılarak bilgi toplanmaktadır. Kategorisi bilinmeyen her alternatif için toplamda m*(q-1) farazi atama gerçekleştirilmektedir. y_{tk}, a_t alternatifinin k kategorisine farazi atanma sayısı olsun. a_t'nin k kategorisine atanma olasılığı Eş. 3.25'te olduğu gibi k kategorisine atanma sıklığının toplam atanma sıklığına bölünmesiyle bulunmaktadır. Burada x_{tk}, a_t alternatifinin k kategorisine atanıp atanmadığı bilgisini temsil etmektedir.

$$p(y_{tk} = 1) = \frac{x_{tk}}{\sum_{r=1}^{q} x_{tr}} \tag{3.25}$$

Model tabanlı atama yaklaşımına ek olarak, alternatiflerin atama sıklığını bulmak için simülasyon tabanlı bir atama yaklaşımı da geliştirilmektedir. Simülasyon tabanlı yaklaşımı uygulamak için KV'nin tercihlerinin parçalı doğrusal bir fonksiyonla tutarlı olduğu varsayılmaktadır. Parametreler için tekdüze dağılım varsayarak Monte Carlo simülasyonları ile 10.000 rastgele örnek seti oluşturulmaktadır. 10.000 uyumlu parametre seti oluşturmak için verimli bir yaklaşım geliştirilmektedir. KV tarafından yeni atamaların eklenmesiyle w_{ip} değerlerinin aralıklarının daralması beklenmektedir. KV kısıtlı sınıflandırma problemlerinde olduğu gibi bu değerler için aralıklar sağlayabilir. Eğer böyle bir bilgi mevcut değilse o zaman w_{ip} değerlerinin aralığını bulmak için DP modelleri çözülmektedir. Bu aralıklar içinde tekdüze bir şekilde dağıldığı varsayılarak w_{ip} değerler üretilerek toplamları bir olacak şekilde ölçeklendirilmektedir.

Kategori daraltma süreçleri KV'ye sorulan alternatifin atanma bilgisi ışığında devam ettiği için KV'ye hangi alternatifin sorulacağı büyük öneme sahiptir. Burada amaç çözüm uzayını en fazla daraltacak alternatifin seçilerek KV'den en az bilgi alınarak karar sürecinin tamamlanmasıdır. KV'ye sorulacak alternatifin seçiminde literatürde farklı yöntemler kullanılmaktadır. Ulu ve Köksalan (2001) KV'nin tercihlerini temsil edebilecek parametreleri bulmak için DP modelleri çözmektedir. Bu modeller sonucunda ortaya çıkan parametrelere göre hesaplanan fayda değerlerinden kategorilerin sınırlarına en yakın alternatif KV'ye sorulmaktadır. Buğdacı vd. (2013) kategori sınırlarından büyük veya küçük olma olasılığı birbirine en yakın olan alternatifi KV'ye sormaktadır.

Yukarıda bahsedilen iki çalışma da alternatiflerin atanma belirsizliklerini tahmin ederek en belirsiz olanını KV'ye sormaktadır. Bu tezde KV'ye sorulacak alternatifi belirlemek için göreceli entropi ölçütü kullanılmaktadır. Shannon (1948) bilgi kuramı bağlamında olasılıksal değişkenlerin taşıdığı belirsizliği ölçmek için entropiyi önermektedir. Eş. 2.26'da entropinin formülü bulunmaktadır. Entropi olasılık değerleri ile bu değerlerin logaritmasının çarpımları toplanarak hesaplamaktadır. Burada logaritma fonksiyonunun düşük olasılıklar için daha yüksek değer vermesi düşük olasılıklı değerlerin daha çok bilgi taşımasıyla doğru orantılıdır. Aynı şekilde olasılığı yüksek bir olayın gerçekleşmesi durumunda daha az bilgi alınmaktadır. Bu şekilde rastgele değişkenlerin taşıdığı belirsizlik ölçülmektedir. Maksimum entropi ise tüm değerlerin aynı olasılığa sahip olduğu durumda gerçekleşmektedir.

$$H(X) = -\sum_{k=1}^{q} p(x_k) log_2\, p(x_k) \qquad (2.26)$$

ÇKS problemleri için entropi yöntemi sıklıkla kullanılmaktadır. Abbas (2004) alternatiflerin von Neumann ve Morgenstern fayda değerlerini tahmin ederek maksimum entropi prensibini kullanarak KV'ye soru sormaktadır. Valkenhoef ve Tervonen (2016) ise Abbas'ın (2004) çalışmasını KV'nin doğrusal bir fayda fonksiyonuna sahip olduğunu varsayarak alternatifleri sıralamak için genişletmektedir. Ciomek vd. (2017) sıralama problemleri için alternatiflerin alabilecekleri sıraları Monte Carlo simülasyonları ile tahmin etmektedir. Sistemin toplam entropisini en çok düşürecek olan ikiliyi KV'ye kıyaslaması için sorarak etkileşimli bir yöntem önermektedir. Wu vd. (2011) ise veri zarflama analizi yöntemiyle sıralama yapmak için entropiyi kullanmaktadır.

Olası değer sayısı farklı olan değişkenlerin belirsizlikleri entropi ile doğru ölçülmeyebilir. Örneğin, dört kategorili bir problemde alternatif 1 ve 2'nin atanabilecekleri olası kategoriler sırasıyla $\{C_1, C_2\}$ ve $\{C_1, C_2, C_3, C_4\}$ olsun. Kategorilere atanma olasılıkları ise sırasıyla {0,50, 0,50} ve {0,05, 0,10, 0,15, 0,70} olsun. Alternatif 1 ve 2'nin entropileri sırasıyla 1 ve 1,32 olarak hesaplanmaktadır. Alternatif 1'in daha yüksek belirsizliğe sahip olması beklenmesine rağmen, alternatif 2'nin olası kategorileri daha fazla olduğu için daha yüksek bir entropiye sahiptir. Shannon (1948) farklı sayıda değerlere sahip değişkenler için göreceli entropiyi önermiştir. Eş. 2.29'da görüldüğü üzere bir alternatifin entropi değeri aynı sayıda olası kategoriye sahip maksimum entropi değerine bölünerek göreceli entropi hesaplanmaktadır. Verilen örnekte alternatiflerin göreceli entropisi sırasıyla 1 ve 0,66 olarak hesaplanmaktadır.

$$H_R(X) = \frac{-\sum_{k=1}^{q} p(x_k) log_2\, p(x_k)}{log_2\, q} \tag{2.29}$$

3.4 Olasılıksal Durum için Algoritma

$DP1_{a_t,k}$ ve $DP2_{a_t,k}$ modelleri kategori aralıklarını daraltarak birçok alternatifin KV'ye sorulmadan doğru kategorisine atanmasını sağlamaktadır. Literatürde tüm alternatifleri doğru kategorisine atayan etkileşimli çalışmalara bakıldığında alternatiflerin yarıya yakını KV tarafından atanmaktadır (örn., Köksalan ve Ulu, 2003; Ulu ve Köksalan, 2014). Bu da KV'nin muhakeme yükünün fazla olmasına sebep olmaktadır. Öte yandan UTADIS gibi etkileşimli olmayan yöntemlerde KV'den toplu olarak birçok referans alternatifin kategori bilgisi alınmaktadır ve DP modeli çözülerek tek bir parametre seti oluşturulup alternatifler bu parametrelere göre sınıflandırılmaktadır. Bu da KV'nin muhakeme yükünü artırırken birçok alternatifin hatalı sınıflandırılmasına sebep olmaktadır. Bu sebeplerle bu makalede etkileşimli ve olasılıksal bir yöntem geliştirerek KV'den alınacak en az bilgi ve en az sınıflandırma hatası ile karar sürecinin tamamlanması hedeflenmektedir.

Bir alternatif için bir kategoriye ait atanma olasılığı diğer kategorilere göre çok daha yüksekse (düşükse) bu alternatifin bu kategoriye ait olma ihtimalinin daha yüksek (düşük) olduğu düşünülebilir. Bu tür alternatiflerin KV'ye sorulmadan atanmasına olasılıksal sınıflandırma denmektedir. Bu çalışmada DP modelleri çözüldükten sonra olasılıklar hesaplanarak olasılıksal sınıflandırma da yapılmaktadır. KV tarafından belirlenmiş bir kritik değer, τ, olasılıksal atamalarda hata payını ifade etmek için tanımlanmaktadır. a_t alternatifi için eğer $p(x_{tk} = 1) \geq 1 - \tau$ ise bu alternatif k kategorisine olasılıksal olarak atanmaktadır. KV'den atama bilgisi almadan olasılıksal atama yapmak, yanlış sınıflandırma oranlarını artıracaktır. Bu sebeple Eş. 2.29'da ifade edildiği üzere alternatiflerin atanma belirsizliklerini ifade eden göreceli entropi değerlerinin kategorisi bilinmeyen alternatifler için ortalaması alınarak sistemin belirsizliği tahmin edilmektedir. KV'den atama bilgisi alındıkça göreceli entropi değerlerinin azalma eğiliminde olması beklenmektedir. Ortalama göreceli entropi (OGE) değeri 0,5'in altına düştüğünde olasılıksal atamalara izin verilmektedir. Aşağıda olasılıksal algoritma verilmektedir. C_0, kategorisi bilinmeyen alternatiflerin kümesini ifade etmektedir.

Algoritma A_{PRENT}

Aşama 1 (*kategori daraltma*): Her $a_t \in C_0$ için $k = C_t^{\text{Eİ}}$ olsun.

1.1. $DP1_{a_t,k}$ modelini çöz.

- Eğer $hf_1^*(a_t, k) \geq 0$ ise $C_t^{\text{EK}} = k$ olarak güncelle. a_t'ye baskın olan her $a_{t'} \in C_0$ için $C_{t'}^{\text{EK}} = k$ olarak güncelle. $C_{t'}^{\text{EK}} = C_{t'}^{\text{Eİ}} = k$ ise $C_k \leftarrow C_k \cup \{a_{t'}\}$ ve $C_0 \leftarrow C_0 - \{a_{t'}\}$ olarak güncelle. $C_t^{\text{EK}} = C_t^{\text{Eİ}} = k$ ise $C_k \leftarrow C_k \cup \{a_t\}$ ve $C_0 \leftarrow C_0 - \{a_t\}$ olarak güncelle ve aşama 1.3'e git. Aksi halde $k = k - 1$ olsun ve aşama 1.2'ye git.
- Eğer $hf_1^*(a_t, k) < 0$ ve $k < C_t^{\text{EK}} - 1$ ise $k = k + 1$ olsun ve aşama 1.1'i tekrar et.
- Eğer $hf_1^*(a_t, k) < 0$ ve $k = C_t^{\text{EK}} - 1$ ise aşama 1.2'ye git.

1.2. $DP2_{a_t,k}$ modelini çöz.

- Eğer $hf_2^*(a_t, k) < 0$ ise $C_t^{\text{Eİ}} = k + 1$ olarak güncelle. a_t'nin baskın olduğu her $a_{t'} \in C_0$ için $C_{t'}^{\text{Eİ}} = k + 1$ olarak güncelle. $C_{t'}^{\text{EK}} = C_{t'}^{\text{Eİ}} = k + 1$ ise $C_k \leftarrow C_k \cup \{a_{t'}\}$ ve $C_0 \leftarrow C_0 - \{a_{t'}\}$ olarak güncelle. $C_t^{\text{EK}} = C_t^{\text{Eİ}} = k + 1$ ise $C_{k+1} \leftarrow C_{k+1} \cup \{a_t\}$ ve $C_0 \leftarrow C_0 - \{a_t\}$ olarak güncelle. Aşama 1.3'e git.
- Eğer $hf_2^*(a_t, k) \geq 0$ ve $k > C_t^{\text{Eİ}}$ ise $k = k - 1$ olarak güncele ve aşama 1.2'yi tekrar et.
- Eğer $hf_2^*(a_t, k) \geq 0$ ve $k = C_t^{\text{Eİ}}$ ise aşama 1.3'e git.

1.3. Eğer C_0'daki tüm alternatifler için kategori daraltma aşaması uygulandıysa, aşama 2'ye git. Aksi halde sıradaki $a_t \in C_0$ ile aşama 1.1'e git.

Aşama 2 (*olasılıksal atama*): Eğer $OGE < 0{,}5$ ise her $a_t \in C_0$ için $p(x_{tk} = 1) \geq 1 - \tau$ olduğunda $C_k \leftarrow C_k \cup \{a_t\}$ ve $C_0 \leftarrow C_0 - \{a_t\}$ olarak güncelle. $C_0 = \emptyset$ ise aşama 4'e git. Aksi halde aşama 3'e git.

Aşama 3 (*KV'ye sorulacak alternatifi belirleme*): KV'ye sormak için $a_t^s \in C_0$ alternatifini seç. a_t^s'nin KV tarafından atandığı kategori $C_{k'}$ olsun. $C_{k'} \leftarrow C_{k'} \cup \{a_t^s\}$ ve $C_0 \leftarrow C_0 - \{a_t^s\}$ olarak güncelle. a_t^s'ye baskın olan her $a_{t'} \in C_0$ için $C_{t'}^{EK} = k'$ olarak; a_t^s'nin baskın olduğu her $a_{t'} \in C_0$ için $C_{t'}^{\text{Eİ}} = k'$ olarak güncelle. Eğer $C_0 = \emptyset$ ise aşama 4'e git. Aksi halde aşama 1'e git.

Aşama 4 (*bitiş*): Tüm alternatiflerin atama bilgilerini KV'ye bildir ve dur.

3.5 Kıyaslama Yapılacak Yaklaşımlar

Bu çalışmada geliştirilen olasılıksız ve olasılıksal algoritmaların sınıflandırma performanslarını ölçmek için literatürden algoritmalar ve KV'ye sorulacak alternatifi seçim yöntemleri kullanılmaktadır. Algoritmaları KV'nin muhakeme yükü açısından kıyaslamak için, ikili kategori kararları (İKK) ölçütü sunulmaktadır. KV tarafından her kategori aralığı daraltması bir İKK olarak kabul edilmektedir. AS_k k olası kategori arasında KV'nin atama sayısı olsun. Bir q-kategori problemindeki toplam İKK sayısı Eş. 4.1'deki gibi tanımlanmaktadır. Örneğin, iki ve üç kategori arasındaki KV atamalarının sayısı sırasıyla α ve β ise, KV'den gereken ikili kategori bilgisi sayısı α+2β olarak hesaplanmaktadır.

$$\dot{I}KK = \sum_{k=2}^{q} (k-1) * AS_k \qquad (4.1)$$

3.5.1 Olasılıksız Durum için Kıyaslama Algoritmaları

İlk kıyaslama algoritması Buğdacı vd. (2013) çalışmasının olasılıksız algoritmasıdır. Yazarlar bir alternatifin faydasının bir kategorinin fayda eşiğinden daha büyük olma olasılığını hesaplamak için DP modelleri kullanmaktadır. Olasılığı 0,5'e en yakın olan alternatif, KV'ye sorulacak en belirsiz alternatif olarak kabul edilmektedir. Burada mantık olarak olasılık değeri 0,5'e eşitse, seçilen alternatifin bir kategorinin eşiğinden daha büyük veya daha küçük bir faydasına sahip olma olasılığı eşit olarak algılanmaktadır.

Olasılıksız durumda kıyaslama yapılacak yöntemler arasında Buğdacı vd. (2013) çalışmasına ek olarak üç alternatif seçim yöntemi kullanılmaktadır. Bu üç yöntem olasılıksız algoritmaya dahil edilerek KV'ye sorulacak alternatifte farklılık göstermektedir. İlki KV tarafından atanacak alternatiflerin rastgele seçildiği rassal algoritmadır. Alternatif seçim sürecinde rassallık olduğu için her bir problemde rastgele 100 örnek oluşturularak ortalamalar raporlanmaktadır. İkinci olarak Özpeynirci vd. (2018) çalışmasındaki alternatif seçim yöntemi kullanılmaktadır. En geniş kategori aralığına sahip alternatifler arasından en yüksek baskınlık ilişkilerine sahip olan alternatif KV'ye sorulmak üzere seçilmektedir. Üçüncü seçim yöntemi, Benabbou vd. (2017) çalışmasının maksimum minimaks pişmanlık yaklaşımıdır.

3.5.2 Olasılıksal Durum için Kıyaslama Algoritması

Bu çalışmada geliştirilen olasılıksal algoritmanın performansını ölçmek için Buğdacı vd. (2013) çalışmasının algoritması (BA) kullanılmaktadır. BA, u_k ve w_{ip} değerlerinin uç değerlerini bulmak için DP modelleri çözmektedir. Bu değerlerin tekdüze dağılım gösterdiği varsayılarak alternatiflerin değerleri ile kategori sınırları arasındaki fark tahmin edilmektedir. Bu farkın normal dağıldığı varsayılarak alternatiflerin değerlerinin kategorilerin sınırlarından büyük olma olasılığı hesaplanmaktadır. τ kritik değerine göre aşağıdaki gibi olasılıksal atama yapılmaktadır.

- Eğer $p(U(a_t) \geq u_1) \geq 1 - \tau$ ise a_t C_1 kategorisine atanmaktadır.
- Eğer $p(U(a_t) \geq u_k) \geq 1 - \tau$ ve $p(U(a_t) \geq u_{k-1}) \leq \tau$ ise a_t C_k kategorisine atanmaktadır.
- Eğer $p\big(U(a_t) \geq u_{q-1}\big) \leq \tau$ ise a_t C_q kategorisine atanmaktadır.

Her iterasyonda KV'nin atama bilgileri kullanılarak DP modelleri yeniden çözülmekte ve olasılıklar güncellenmektedir. Olasılıksal atama yapıldıktan sonra KV'den bir alternatifin kategori bilgisi istenmektedir. Olasılık değeri 0,5'e en yakın olan alternatif, KV'ye sormak için en belirsiz alternatif olarak kabul edilmektedir. 0,5'e en yakın olasılığa sahip alternatifin seçilmesindeki amaç iki kategori arasında atanma belirsizliği en fazla olanı seçmektir.

4. Uygulamalar ve Sonuçlar

Geliştirilen olasılıksız ve olasılıksal algoritmalar ve kıyaslama algoritmaları literatürden üç örnek problem üzerinde ve ek olarak rastgele oluşturulmuş problemler üzerinde uygulanmaktadır. Birinci problemde üç kriterle değerlendirilen 81 adet MBA programının üç kategoriye ayrılması istenmektedir. İkinci problemde ise sekiz kriterle değerlendirilen 76 adet otobüsün tamire ihtiyaç duyulup duyulmadığına dair dört kategoriye ayrılması istenmektedir. Üçüncü problemde ise dört kriterle değerlendirilen 128 ülkenin enerji performansları açısından dört kategoriye ayrılması istenmektedir. Son olarak dördüncü problemde rassal olarak üretilen veri ile problemler oluşturulmaktadır. Weibull dağılımı kullanılarak farklı parametre değerleri ile her durum için 100 farklı veri seti oluşturulmaktadır. Üç kriterle değerlendirilen 100 alternatifin beş kategoriye ayrılması beklenmektedir.

İlk iki problem için kısıtsız ve kısıtlı durumlarda olasılıksız algoritmalar uygulanmaktadır. Üçüncü problemde ve rastgele oluşturulmuş problemlerde olasılıksız ve olasılıksal algoritmalar birlikte ele alınmaktadır. Birinci ve üçüncü problemlerde parçalı doğrusal tercih fonksiyonları varsayılırken ikinci problemde genel monoton tercih fonksiyonları ele alınmaktadır. Rassal oluşturulmuş problemlerde her iki toplamsal fonksiyon da dâhil edilmektedir.

Sonuçlara göre KV'den atama yapması istendiğinde geliştirilen algoritmaların daha az kategori arasından seçim yaptırma eğiliminde olduğu görülmektedir. Geliştirilen algoritmalar KV'den elde edilen atama bilgileri ve çözülen model sayısı açısından kıyaslama algoritmalarından daha iyi performans göstermektedir. Daha geniş kategori aralıklarına sahip alternatiflerin atanmasının sisteme daha değerli bilgiler getirmesi beklenebilir. Ancak bu çalışmada kategori aralıklarının atamaların belirsizliğini temsil etmediğini ve göreli entropi tabanlı yöntemin alternatiflerin belirsizliğini belirlemede iyi çalıştığı gösterilmektedir.

Geliştirilen olasılıksal algoritmaların uygulama sonuçlarına göre KV'den yeterli bilgi alınmadan olasılıksal atama yapmama kuralının kıyaslama yapılan $A_{PBUĞ}$ algoritmasına göre daha az sayıda yanlış sınıflandırmaya neden olduğu görülmektedir. Ayrıca, en düşük göreceli entropiye sahip alternatiflerin olasılıksal olarak atandığı yaklaşımda KV'den elde edilen bilgi sayısının yüksek ve yanlış sınıflandırma sayısının düşük olduğu ve küçük τ değerleri durumu ile benzer bir atama performansı verdiği gözlemlenmektedir.

Model tabanlı ve simülasyon tabanlı varsayımsal atamalara sahip algoritmalar, KV'den elde edilen atama bilgileri ve olasılıksal atamaların doğruluğu açısından benzer performanslar göstermektedir. Simülasyona dayalı teknik, KV yeterli sayıda alternatif atadığında 10.000 uyumlu parametre seti oluşturmak için çok fazla zaman almaktadır. KV'nin tercihlerinin genel bir monoton tercih fonksiyonu ile tutarlı olduğu varsayıldığı etkileşimli bir ortamda simülasyon tabanlı bir yaklaşım yürütmenin mümkün olmadığı ortaya çıkmaktadır.

Çalışmanın iki kısıtlaması bulunmaktadır. İlk olarak, kategori, kriter veya alt aralık sayısı arttıkça olasılıksız durumda KV'nin bilişsel yükü de artmaktadır. Örneğin, 5 kategori, 3 kriter ve 100 alternatifli rassal veri uygulamasında KV tarafından yaklaşık 70 atama yapılması beklenirken matematiksel modeler tarafından 30 atama yapılmaktadır. Ayrıca olasılıklı durumda kategori sayısı arttıkça yanlış sınıflandırmaların sayısı da artmaktadır. Gelecekteki bir araştırma olarak KV'nin muhakeme yükünü ve yanlış sınıflandırmaları azaltmak için farklı durma koşulları denenebilir.

İkinci kısıtlama olarak bu çalışmada kullanılan modellerin KV'nin tercih fonksiyonu ile tutarlı olduğu varsayılarak matematiksel modellerin her zaman olurlu bir çözüm sunmasıdır. Ancak, KV'nin atamaları toplamsal fayda fonksiyonları ile tutarlı olmayabilir. Matematiksel modeller olursuz çözümlerle sonuçlanarak bu tür tutarsızlıkları tespit edecektir. Tutarsızlıkları ele almanın bir yolu, tercihlerini gözden geçirmesi için KV'ye sunmaktır. Ek olarak, Ciomek vd. (2017) çalışmasındaki gibi KV'nin sürecin önceki aşamalarına geri dönmesi sağlanabilir. Tutarsızlıkları ele almanın bir başka yolu da matematiksel modellerin olursuzluğunu ele almaktır. Sınıflandırma hatalarının toplamını en aza indirmek, maksimum sınıflandırma hatasını en aza indirmek ve yanlış sınıflandırma sayısını en aza indirmek gibi amaç fonksiyonları kullanılabilir (örn., Chinneck, 2008).

Geliştirilen algoritmalar lisansüstü öğrenci kabulünün yanı sıra hasta veya tedarikçi sınıflandırması gibi birçok probleme doğrudan uygulanabilir. Gelecekteki bir araştırma yönü olarak, monoton olmayan kriterlerle ÇKS problemlerine etkileşimli bir yaklaşım geliştirmek için bu çalışma genişletilebilir. Diğer araştırma alanları, atamaların olasılıklarını belirlemek için Bayes yaklaşımı kullanmak ve yarı-içbükey veya L_p norm fonksiyonları gibi farklı fayda fonksiyonu formlarıyla çalışmak olabilir.

C. THESIS PERMISSION FORM / TEZ İZİN FORMU

(Please fill out this form on computer. Double click on the boxes to fill them)

ENSTİTÜ / INSTITUTE

Fen Bilimleri Enstitüsü / Graduate School of Natural and Applied Sciences	☐
Sosyal Bilimler Enstitüsü / Graduate School of Social Sciences	☒
Uygulamalı Matematik Enstitüsü / Graduate School of Applied Mathematics	☐
Enformatik Enstitüsü / Graduate School of Informatics	☐
Deniz Bilimleri Enstitüsü / Graduate School of Marine Sciences	☐

YAZARIN / AUTHOR

Soyadı / Surname : Özarslan
Adı / Name : Ali
Bölümü / Department : İşletme / Business Administration

TEZİN ADI / TITLE OF THE THESIS (**İngilizce** / English): Information Theoretic Measure Based Interactive Approaches to Multi-Criteria Sorting Problems

TEZİN TÜRÜ / DEGREE: **Yüksek Lisans** / Master ☐ **Doktora** / PhD ☒

1. **Tezin tamamı dünya çapında erişime açılacaktır. /** Release the entire work immediately for access worldwide. ☐
2. **Tez iki yıl süreyle erişime kapalı olacaktır.** / Secure the entire work for patent and/or proprietary purposes for a period of **two years**. * ☐
3. **Tez altı ay süreyle erişime kapalı olacaktır.** / Secure the entire work for period of **six months**. * ☒

***** *Enstitü Yönetim Kurulu kararının basılı kopyası tezle birlikte kütüphaneye teslim edilecektir. / A copy of the decision of the Institute Administrative Committee will be delivered to the library together with the printed thesis.*

Yazarın imzası / Signature .... **Tarih** / Date ....
(Kütüphaneye teslim ettiğiniz tarih. Elle doldurulacaktır.)
(Library submission date. Please fill out by hand.)

Tezin son sayfasıdır. / This is the last page of the thesis/dissertation.