



UNIVERSITY OF LEEDS

This is a repository copy of *Supporting the Multi-Criteria Decision Aiding process: R and the MCDA package*.

White Rose Research Online URL for this paper:
<http://eprints.whiterose.ac.uk/112838/>

Version: Accepted Version

Article:

Bigaret, S, Hodgett, RE orcid.org/0000-0002-4351-7240, Meyer, P et al. (2 more authors) (2017) Supporting the Multi-Criteria Decision Aiding process: R and the MCDA package. EURO Journal on Decision Processes, 5 (1-4). pp. 169-194. ISSN 2193-9438

<https://doi.org/10.1007/s40070-017-0064-1>

© Springer-Verlag Berlin Heidelberg and EURO - The Association of European Operational Research Societies 2017. This is an author produced version of a paper published in EURO Journal on Decision Processes. The final publication is available at Springer via <https://doi.org/10.1007/s4007>. Uploaded in accordance with the publisher's self-archiving policy.

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Supporting the Multi-Criteria Decision Aiding process: R and the MCDA package

Sébastien Bigaret · Richard E. Hodgett ·
Patrick Meyer · Tatiana Mironova ·
Alexandru-Liviu Olteanu

the date of receipt and acceptance should be inserted later

Abstract Reaching a decision when multiple, possibly conflicting, criteria are taken into account is often a difficult task. This normally requires the intervention of an analyst to aid the decision maker in following a clear methodology with respect to the steps that need to be taken, as well as the use of different algorithms and software tools. Most of these tools focus on one or a small number of algorithms, some are difficult to adapt and interface with other tools, while only a few belong to dynamic communities of contributors allowing them to expand in use and functionality. In this paper, we address these issues by proposing to use the R statistical environment and the MCDA package of decision aiding algorithms and tools. This package is meant to provide a wide range of MCDA algorithms that may be used by an analyst to tailor a decision aiding process to their needs, while the choice of R takes advantage of the yet poorly explored opportunity to interface data analysis and decision aiding. We additionally demonstrate the use of this tool on a practical application following a well-defined decision aiding process.

Keywords R · MCDA · decision aiding process

1 Introduction

Over the past 50 years, many articles and books have covered the topic of Multi-Criteria Decision Aiding (MCDA) with many different methods and algorithms being proposed. The interested reader can for example refer to [Roy \(1991\)](#);

Sébastien Bigaret · Patrick Meyer · Tatiana Mironova · Alexandru-Liviu Olteanu
IMT Atlantique, Technopôle Brest Iroise, CS 83818, 29238 Brest Cedex 3, France
UMR CNRS 6285 Lab-STICC

Richard E. Hodgett
Leeds University Business School, Maurice Keyworth Building, The University of Leeds, Leeds,
LS2 9JT, United Kingdom

Patrick Meyer
Corresponding author
Tel.: +33 (0)2 29 00 14 63, Fax.: +33 (0)2 29 00 10 30
E-mail: patrick.meyer@imt-atlantique.fr

5 [Bouyssou et al \(2006\)](#); [Belton and Stewart \(2002\)](#). Within the MCDA framework
6 we generally identify at least one decision-maker (DM), who is in charge of and
7 responsible for the decision to be made. He is confronted with several decision alter-
8 natives which are evaluated on a set of criteria or points of view, which typically
9 are conflicting. The DM usually expresses some preferences related to these alter-
10 natives and criteria, which are usually used as parameters by MCDA algorithms
11 in order to provide a solution to the decision problem. The decision problem may
12 also fall into different categories, as e.g., choice (determine the “best” alternative),
13 ranking (order the alternatives from the “best” to the “worst” one) and sorting
14 (assign the alternatives to predefined and ordered classes). In order to illustrate
15 these concepts, let us present a short example. A school committee is tasked with
16 allocating a fixed number of scholarships to students based on their performances
17 on the subjects they are being taught (e.g. mathematics, computer science, bi-
18 ology, etc.). In this case, the school committee is the DM, the students are the
19 decision alternatives, while the subjects represent the criteria. The decision prob-
20 lem, in this case, is to rank all students from best to worst (ranking problem) and
21 to select the top students as recipients of a scholarship. This ranking has to be
22 done according to the preferences of the school committee.

23 MCDA has been applied to many different fields, such as health ([Wahlster
24 et al, 2015](#)), finance and banking ([Figueira et al, 2005](#), p. 799), environmental man-
25 agement ([Lahdelma et al, 2014](#)), urban planning using geographical information
26 systems ([Coutinho-Rodrigues et al, 2011](#)), robotics ([Taillandier and Stinckwich,
27 2011](#)), energy planning ([Figueira et al, 2005](#), p. 859), nuclear emergency manage-
28 ment ([Papamichail and French, 2013](#)), equipment selection ([Hodgett, 2016](#)) etc.
29 The process of decision aiding is often complex, depending on the specific field of
30 application and the preferences of the DM. As a result, many MCDA algorithms
31 have been developed over the years (see for example ([Figueira et al, 2005](#); [Keeney
32 and Raiffa, 1976](#))). In order to overcome the difficulties linked to the decision
33 problem, an analyst may be included in the decision aiding process. (S)he is an
34 expert of MCDA, whose purpose is to guide the DM by choosing the correct for-
35 malization of the problem, the appropriate methods and algorithmic approaches,
36 in order to support him/her in reaching a decision recommendation. In order to
37 simplify and streamline the decision aiding process, several studies have already
38 dealt with the topic of selecting the best suited algorithm for a decision problem
39 ([Guitouni et al, 1998](#); [Ishizaka and Nemery, 2013](#)), while others, as, e.g., [Tsoukias
40 \(2007\)](#), have divided this process into multiple steps. Many software solutions have
41 been proposed to help the analyst in the decision aiding process, however, in most
42 cases they hold several limitations. Plenty of them focus on only a small number of
43 algorithms, raising the need to use multiple software tools throughout the decision
44 aiding process and the potential difficulties linked to their coupling. Other tools
45 limit the capacity of the user to adapt their algorithms to their needs, while only
46 a few belong to dynamic communities of contributors allowing them to grow in
47 use and functionality.

48 These remarks provide the key motivation for this contribution. The MCDA
49 package ([Meyer et al, 2015](#)) for the R statistical software ([R Development Core
50 Team, 2008](#)) that we propose is meant to provide a wide range of algorithms that
51 may be used by an analyst across an entire decision aiding process. The choice
52 of R is also motivated by the ease in adapting the different functions to one’s
53 needs, the large community of contributors that may aid in extending the MCDA

54 package, as well as the as of yet poorly explored opportunity of interfacing data
55 analysis and decision aiding. Both the data analysis community may benefit from
56 the possibility of applying decision aiding algorithms after the data analysis stage,
57 as well as the decision aiding community from the possibility of applying data
58 analysis during the decision aiding process.

59 The remainder of this article is organized in the following way. In Section 2
60 we provide a state of the art, starting with the MCDA process, the different algo-
61 rithms that have been proposed and finishing with an overview of the most notable
62 supporting software tools. In Section 3 we present and discuss our proposal to use
63 the R statistical environment combined with our contribution, the MCDA package
64 for R. In Section 4 we provide a very detailed illustrative example showcasing how
65 R and the package may be used in practice. Finally, in Section 5 we finish with
66 several conclusions and perspectives for future work.

67 2 State of the art

68 We start by providing a state of the art on the existing MCDA literature, cov-
69 ering the decision aiding process, the most commonly used algorithmic approaches,
70 as well as some of the existing supporting software tools.

71 2.1 The multi-criteria decision aiding process

72 As mentioned in the introduction, decisions and the objects they are concerned
73 with may be very diverse. In fact, each of us are faced with a multitude of decisions
74 every day, from which route to take in order to get to work in the morning,
75 to selecting what to have for lunch. There are numerous factors which influence
76 these decisions, such as our preferences, our prior experiences, different constraints,
77 etc. In certain cases, balancing these factors can be difficult. MCDA serves as an
78 interface between DMs and analysts, guiding them in reaching a decision when
79 multiple and often conflicting criteria are involved. The process generally starts
80 with the analyst and DMs focusing on defining the problem, their goals and how
81 the final decision should be reached (Bouyssou et al, 2006). One key aspect of
82 MCDA is that the final decision may not need to be the best possible one, but one
83 that is acceptable by all the stakeholders. Hence, when multiple DMs are involved,
84 conflicts need to be handled in order to reach a consensus on the final decision.
85 The term “decision frame”, used by Tversky and Kahneman (1981), supports the
86 fact that DMs often base their decision on subjective judgments. Furthermore,
87 nowadays real-world decision problems have become increasingly complex.

88 The following steps have been identified to structure a MCDA process: identi-
89 fy the problem, formulate the problem, construct the evaluation model and then
90 reach a final recommendation (Bouyssou et al, 2006; Bisdorff et al, 2015; Figueira
91 et al, 2005). Each of these steps contains additional sub-steps, which depend on
92 the many factors that define a decision problem. Figure 1 illustrates an example
93 decision aiding process and the complexity of the various steps involved. More pre-
94 cisely, the step of structuring the problem includes sub-steps such as identifying
95 the stakeholders (or actors), identifying the context of the problem, the objectives
96 of the decision and its respective properties. The second step of formulating the

97 problem, involves identifying the decision alternatives and their criteria, the type
 98 of decision problem, as well as managing multiple DMs and their different perspec-
 99 tives. The third step involves the choice of a mathematical model and its tuning so
 100 that it reflects the perspective of the DM. Furthermore, a resolution method also
 101 needs to be selected in order to provide a recommendation to the decision problem.
 102 Finally, in the last step, this recommendation is presented to the DM, who then
 103 either validates the recommendation, asks for additional supporting analyses or
 104 revisits previous steps in order to refine the solution. We would like to highlight
 105 that the structure of the process is nonlinear, complex and iterative. This means
 106 that any point in the process we may choose different paths to follow, in some
 107 cases going back to previous ones. We highlight this, and the fact that the deci-
 108 sion aiding process is decomposed into multiple sub-steps, as shown in Figure 1. At
 109 each of these steps we may have an interaction between the DM and the analyst,
 110 the extraction of an important piece of information, the use of an algorithm, or a
 111 visual representation of alternative, etc.

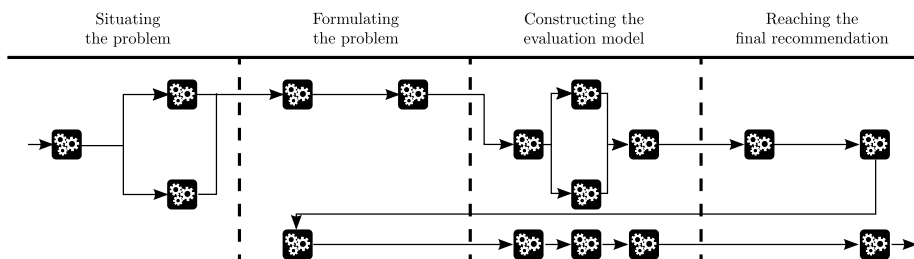


Fig. 1 The decision aiding process - example of one possible instance.

112 2.2 MC aggregation algorithms

113 The resolution step of the previously described MCDA process involves the
 114 use of an MC aggregation algorithm. Various such algorithmic approaches have
 115 been proposed in the literature (Bouyssou et al, 2006; Keeney and Raiffa, 1976;
 116 Roy, 1996). Roughly speaking, two main methodological schools can be identified,
 117 namely the outranking school of thought and the value-based theories.

118 The main idea behind *outranking methods* is to compare any two alternatives
 119 pair-wise on basis of their evaluations on the set of criteria, according to a majority
 120 rule. For two alternatives x and y of A , if for the DM there are enough arguments in
 121 favor of the statement “ x is at least as good as y ”, then x outranks y (xSy) (Roy,
 122 1996). These arguments are based on differences of evaluations on the various
 123 criteria which are compared to discrimination thresholds determined in accordance
 124 with the DM’s preferences. Furthermore, a weight is associated with each criterion,
 125 which allows giving these local arguments more or less importance in the majority
 126 rule. A concordance index then aggregates these partial arguments via a weighted
 127 sum to obtain a credibility degree of the outranking. Three preference situations
 128 can be derived from this outranking relation. x and y are considered as indifferent if
 129 simultaneously xSy and ySx , they are considered as incomparable with respect to

130 the available information if no outranking can be confirmed between them (neither
131 xSy nor ySx), and x (resp. y) is strictly preferred to y (resp. x) if xSy and not ySx
132 (resp. ySx and not xSy). As this outranking relation is not necessarily complete
133 or transitive, its exploitation in view of building a decision recommendation is
134 in general quite difficult. Many exploitation procedures have been proposed in
135 the literature to solve the three main types of multi-criteria decision problems
136 mentioned in Section 2.1.

137 Methods based on *multiattribute value theory* aim to construct a numerical rep-
138 resentation of the DM's preferences on the set of alternatives A . More formally,
139 those techniques seek at modeling the preferences of the DM, supposed to be a
140 weak order represented by the binary relation \succsim on A , by means of an overall value
141 function $U : A \rightarrow \mathbb{R}$ such that $x \succsim y \iff U(x) \geq U(y), \forall x, y \in A$. The overall value
142 function U can be determined via many different methods, presented for example
143 in von Winterfeldt and Edwards (1986, Chapter 8) in the context of an additive
144 value function model. Ideally, such methods should consist of a discussion with
145 the DM in the language of his/her expertise, and avoid technical questions linked
146 to the model which is used. Note that the preference relation induced by such an
147 overall value function is necessarily a complete weak order, which means that only
148 two preference situations can occur : either x and y are considered indifferent (if
149 $U(x) = U(y)$) or x (resp. y) is strictly preferred to y (resp. x) if $U(x) > U(y)$ (resp.
150 $U(y) > U(x)$). Once the overall value function has been properly determined, its
151 exploitation for the decision recommendation is usually straightforward, as all the
152 alternatives have become comparable.

153 The main differences between these two methodological schools lie in the way
154 the alternatives are compared and in the type of information which is required
155 from the decision maker. Furthermore, outranking methods might be preferable
156 if the evaluations of the alternatives on the criteria are mainly qualitative and if
157 the DM would like to include some impreciseness about his/her preferences in the
158 model, whereas value-based methods can be favored if a compensatory behavior
159 of the DM should be modeled.

160 2.3 MCDA software

161 As we have previously discussed, many MCDA methods have been proposed in
162 order to solve different types of decision problems. In order to help applying these
163 methods to real decision problems, a wide range of software have been developed.
164 Some of these software packages are either free (as in beer or as in speech) or only
165 commercially available, while some of them are either stand-alone or web-based.
166 Some software allow to be extended and therefore also gather a community of
167 developers around them. Next to that, many of these software tools provide only a
168 limited number of algorithms, in some cases only single methods (e.g. IRIS by Dias
169 and Mousseau (2003), ELECTRE TRI by Mousseau et al (1999, 2000), MakeItRa-
170 tional by Make It Rational (2016), TransparentChoice by TransparentChoice Ltd.
171 (2016), TOPSIS by Statistical Design Institute (2016), UTA Plus by Kostkowski
172 and Slowinski (1996), JSMAA by Tervonen (2012)), while in other cases multi-
173 ple methods (e.g. the diviz ecosystem with the XMCDA web-services (Meyer and
174 Bigaret, 2012a) developed under the impetus of the Decision Deck Consortium,
175 or Decernes MCDA by Yatsalo et al (2015)). For a broader review of the existing

176 software tools, the reader may refer to [Mustajoki and Marttunen \(2013\)](#); [Ishizaka](#)
177 [and Nemery \(2013\)](#); [Weistroffer et al \(2005\)](#); [Mayag et al \(2011\)](#); [Baizylidayeva et al](#)
178 [\(2013\)](#) and [International Society on Multiple Criteria Decision Making \(2014\)](#).

179 We notice from these reviews of MCDA software, that no software tool is
180 currently able to support the entire complex decision aiding process from start to
181 finish. Additionally, according to [Clemen and Reilly \(2001\)](#), decomposition plays
182 a crucial role in the decision process, while multi-method platforms appear to
183 be more useful due to the possibility of choosing among different algorithms for
184 solving the same problem. There are, nevertheless tools that allow for a wide
185 degree of tuning of the methods they implement, such as for example *diviz* ([Meyer](#)
186 [and Bigaret, 2012a](#)). The *diviz* workbench provides an interface for constructing
187 complex MCDA algorithms from smaller components (available as the XMCDAs
188 web-services of the Decision Deck Consortium), which can be interconnected in
189 the form of work-flows. In line with this notion of being able to tailor different
190 methods and tools to one's needs are the R statistical environment [R Development](#)
191 [Core Team \(2008\)](#) and the Python programming language. In both cases, we find
192 some of the fastest growing communities of contributors and the ability to easily
193 interconnect their contributions in order to solve different problems ([Piatetsky,](#)
194 [2016](#)). R, in particular, is widely used in the data science discipline, where we find
195 a significant opportunity of adding MCDA approaches to be used after the data
196 analysis stages. Similarly, integrating MCDA and data analysis can reveal new
197 challenges for the MCDA community.

198 It should be nevertheless noted that R packages linked to MCDA methods or
199 that may be used in conjunction with them in the decision aiding process also
200 exist: *Kappalab* by [Grabisch et al \(2006, 2015\)](#), *RXMCDAs* by [Meyer and Bigaret](#)
201 [\(2012b\)](#), *UTAR* by [Leistedt \(2011\)](#), *Benchmarking* by [Bogetoft and Otto \(2015\)](#)
202 or *Rgraphiz* by ([Gentry et al, 2009, 2016](#)).

203 All of these remarks serve as motivation for our proposal of the MCDA R
204 package. Our aim is to provide as many different MCDA methods and tools as
205 possible and to make them available to the R and the data analysis communities.
206 In line with the philosophy of R, the package will allow the analyst to construct
207 their own decision aiding process from start to finish, by applying the methods
208 provided by the package, adapting them to their needs as well as making use of
209 other methods and packages linked to data analysis. The functions of the MCDA R
210 package are also easily integrable in the XMCDAs web-services proposal of Decision
211 Deck, and consequently in the *diviz* workbench. Last but not least, we hope that
212 by proposing a library of MCDA functions in an environment like R will create
213 a community of contributors which will participate in its dissemination and the
214 general development effort.

215 **3 R and the MCDA package**

216 We present, in this section, our contribution, by first giving a brief presentation
217 of the philosophy behind R and the package of MCDA functions that we propose,
218 followed by a slightly more in-depth description of the contents of the package,
219 namely the currently implemented functions.

220 3.1 Philosophy

221 R is an open-source functional programming language and environment mainly
222 centered around data analysis (Venables et al, 1998; Ihaka and Gentleman, 1996).
223 In recent years it has grown in popularity with the IEEE identifying it as the 9th
224 most popular programming language in 2014, the 6th most popular in 2015 and
225 the 5th most popular in 2016 (IEEE Spectrum, 2016). Due to the large commu-
226 nity of R users, many tools in the form of functions within packages have been
227 proposed, many dealing with handling different data formats, data pre-processing,
228 data filtering and interactive visualizations. Although users need to have some ba-
229 sic programming experience they also first need to familiarize themselves with the
230 R programming language. Once this is done, however, users can easily combine
231 functions from different packages in order to solve their problem. Nevertheless, the
232 majority of functions and packages are aimed at data analysis, and while there are
233 a few packages linked to MCDA, there is plenty still to be done in this regard.

234 The MCDA package that we propose follows the philosophy of R, by encom-
235 passing a growing array of MCDA algorithms that may be used to decompose
236 the decision aiding process into sub-steps. The package mainly targets MCDA
237 practitioners that are familiar with the decision aiding process, giving them the
238 possibility to construct this process as they see fit. As very often during a decision
239 aiding process the DM does not have a clear picture of his/her problem (Simon,
240 1976), being able to quickly adapt the process as new information is made available
241 is of great importance. Finally, the MCDA package may benefit both MCDA prac-
242 titioners and data analysts, as MCDA practitioners could further apply methods
243 linked to data analysis throughout the decision aiding process, while data analysts
244 could use their data for reaching an objective in addition to analyzing it.

245 3.2 Currently implemented functions

246 At the time of writing, the package is very young and consequently is far
247 from covering all of the algorithms from the classical MCDA literature. However,
248 functions supporting various steps of the MCDA process have been implemented
249 in the MCDA R package. They can be categorized as follows :

- 250 – state of the art aggregation algorithms;
- 251 – state of the art preference elicitation algorithms;
- 252 – tool and data manipulation functions;
- 253 – plot functions.

254 The implemented algorithms originate from the two main methodological schools
255 presented in Section 2.2.

256 With respect to the *aggregation algorithms*, in the outranking paradigm, the
257 currently implemented functions focus on a majority-rule sorting technique com-
258 monly called MR-Sort (Leroy et al, 2011; Sobrie et al, 2013), which is a sim-
259 plified version of the classical Electre TRI method. The `MRSort` function allows
260 to assign alternatives to a set of predefined categories according to a DM's pref-
261 erences. This method has recently been extended to take into account so-called
262 large performance differences by Meyer and Olteanu (2017). This extension is im-
263 plemented in the `LPDMRSort` function. Concerning multiattribute value theory,

264 the aggregation can be done with a weighted sum through the `weightedSum`
265 function, which calculates the weighted sum of the evaluations of alternatives on
266 criteria with respect to some criteria weights. To apply piece-wise linear value func-
267 tions on a performance table, the `applyPiecewiseLinearValueFunctions-`
268 `OnPerformanceTable` can be used. The package also proposes to use the AHP
269 function, which implements the Analytic Hierarchy Process proposed by Saaty
270 (1980), as well as the `pairwiseConsistencyMeasures` function which calcu-
271 lates four pairwise consistency checks for AHP (Siraj et al, 2015). Furthermore,
272 the package proposes an implementation of the TOPSIS method originally pro-
273 posed by Hwang and Yoon (1981) (`TOPSIS` function) and the MARE method by
274 Hodgett et al (2014) (`MARE` function)

275 In terms of *preference elicitation algorithms*, in the outranking school, the param-
276 eters for the MR-Sort technique can be learned from assignment examples provided
277 by the DM, either via the `MRSortInferenceExact` function (exact elicitation
278 via linear programming), or the `MRSortInferenceApprox` function (approximate
279 elicitation, adapted for large sets of assignment examples). The `MRSort-`
280 `IdentifyIncompatibleAssignments` function can be used to identify assign-
281 ment examples which are not compatible with an MR-Sort model. In a context of
282 large performance differences, the `LPDMRSortInferenceExact` function allows
283 to learn the preferential parameters from assignment examples. In case some as-
284 signments are incompatible with the large performance differences sorting model,
285 they can be found via the `LPDMRSortIdentifyIncompatibleAssignments`
286 function. Concerning multiattribute value theory, the package currently proposes
287 preference elicitation methods related to the UTA technique originally proposed
288 by Jacquet-Lagrèze and Siskos (1982). The UTA and UTASTAR functions allow
289 to learn piece-wise linear value functions from rankings of alternatives, whereas
290 the UTADIS function identifies such value functions together with category limits
291 from assignment examples. The `additiveValueFunctionElicitation` func-
292 tion elicits a general additive value function from a ranking of alternatives.

293 Next to these algorithms which represent the heart of the MCDA process, the
294 package provides some *tool and data manipulation* functions. Evaluations in a per-
295 formance table can be normalized according to various normalization schemes in
296 function `normalizePerformanceTable`. Alternatives can be assigned to cate-
297 gories with respect to some separation thresholds via the `assignAlternatives-`
298 `ToCategoriesByThresholds` function.

299 Finally, to show the DM results or intermediary elements of the decision aiding
300 process, a certain number of *plot* functions have been implemented. `plotRadar-`
301 `PerformanceTable` allows to represent the alternatives very synthetically as
302 radar plots. In the outranking context, `plotMRSortSortingProblem` plots the
303 profiles of the alternatives and the categories for a sorting problem. In multiat-
304 tribute value theory, `plotPiecewiseLinearValueFunctions` can be used to
305 plot the piece-wise linear value functions (learned for example from a UTA-like
306 method), whereas `plotAlternativesValuesPreorder` shows the pre-order of
307 the alternatives obtained from their overall scores. Finally, the `plotMare` function
308 presents a synthetic vision of the output of the Mare method.

309 The work on the package is ongoing, and we encourage the interested reader
310 to contribute to this collective effort.

311 4 Illustrative example

312 In this section we present the use of the MCDA R package on a didactic MCDA
 313 problem which has been widely discussed in the literature, namely the choice of a
 314 sports car (see Bouyssou et al (2000), Chapter 6). We show how the package can
 315 be used in the various steps of the MCDA process, which was described in (2.1).
 316 In a real-world decision aiding process, there might be round-trips between these
 317 different steps, in order, for example, to tune the input and output parameters of
 318 the various algorithms.

319 4.1 Problem description

320 This example is inspired from Chapter 6 of Bouyssou et al (2000), but in
 321 order to illustrate all the steps which we wish to highlight, we take the liberty of
 322 slightly modifying the original description. As an illustration of the step "situating
 323 the problem" we have the following information. The problem takes place in 1993,
 324 when Thierry, a student aged 21, is passionate about sports cars and wishes to buy
 325 a middle range, 4 years old car with a powerful engine. He asks an analyst to help
 326 him to find the best alternative for his needs. We will play the role of the analyst
 327 in this decision aiding process. In a first step, we identify the alternatives and the
 328 criteria in a dialogue with Thierry. Three points of view appear to be important
 329 to Thierry, which are expressed through five criteria: cost point of view (criterion
 330 $g1$), performance of the engine point of view (criteria $g2$ and $g3$) and safety point of
 331 view (criteria $g4$ and $g5$). The list of alternatives and their evaluations on these five
 332 criteria is presented in Table 1. Thierry is then asked to express the preferential
 333 direction on each of the criteria. He considers that the "cost" criterion (€) and
 334 the performance criteria "acceleration" (seconds) and "pick up" (seconds) have
 335 to be minimized, whereas the safety criteria "brakes" and "road-hold" have to be
 336 maximized. The values of the latter two criteria are average evaluations obtained
 337 from multiple qualitative evaluations which have been re-coded as integers between
 338 0 and 4. Further details on these data can be found in Bouyssou et al (2000),
 339 Chapter 6. Note here that, in comparison to Bouyssou et al (2000) we removed
 340 a10 on purpose from these data, as it will be used later in our decision aiding
 341 scenario.

342 The initial meeting between Thierry and the analyst, as well as the session of
 343 identifying the decision alternatives, the criteria on which they are defined and
 344 the decision problem correspond each to one activity within the decision aiding
 345 process. The first activity is contained within the first step of situating the problem,
 346 while the second is contained within the second step of formulating the problem.
 347 We illustrate these steps within the decision aiding process through 1.a and 1.b in
 348 Figure 2.

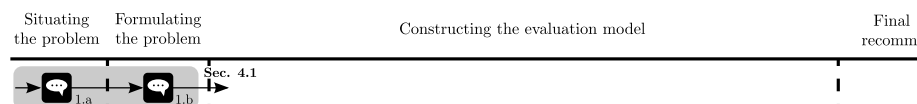


Fig. 2 First part of Thierry's decision aiding process.

Table 1 Data for Thierry's car selection problem.

car ID	car name	cost ($g1$, €)	accel. ($g2$, s)	pick up ($g3$, s)	brakes ($g4$)	road-holding ($g5$)
a01	Tipo	18342	30.7	37.2	2.33	3
a02	Alfa	15335	30.2	41.6	2	2.5
a03	Sunny	16973	29	34.9	2.66	2.5
a04	Mazda	15460	30.4	35.8	1.66	1.5
a05	Colt	15131	29.7	35.6	1.66	1.75
a06	Corolla	13841	30.8	36.5	1.33	2
a07	Civic	18971	28	35.6	2.33	2
a08	Astra	18319	28.9	35.3	1.66	2
a09	Escort	19800	29.4	34.7	2	1.75
a11	P309-16	17537	28.3	34.8	2.33	2.75
a12	P309	15980	29.6	35.3	2.33	2.75
a13	Galant	17219	30.2	36.9	1.66	1.25
a14	R21t	21334	28.9	36.7	2	2.25

349 4.2 Use of the MCDA R package to support the decision aiding process

350 Below, we continue by illustrating the use of R and the MCDA package through-
 351 out the rest of the decision aiding process. We will divide the discourse further
 352 based on the type of evaluation model that will be used. Note also that a file
 353 containing the code which we detail step by step hereafter can be found in the
 354 directory of the package after its installation. To retrieve the path, the following
 355 code can be used:

```
356 # path to the R script of the example
357
358 system.file("examples", "articleExample.R", package="MCDA")
```

359 4.2.1 Filtering rules

360 First of all, the performances of the cars on the various criteria are loaded into
 361 an R data frame. To achieve this, the following code is used:

```
362 # load performance table csv file
363 # provided with the MCDA package
364
365 f <- system.file("datasets", "performanceTable2.csv", package="MCDA")
366
367 pT <- read.csv(file = f, header=TRUE, row.names=1)
```

368 Thierry first wishes to set some rules on the evaluations in order to filter out
 369 certain cars. Consequently he asks that only cars respecting the following set of
 370 constraints are kept:

$$\begin{aligned}
 & \text{brakes } (g4) \geq 2 \\
 & \text{road-hold } (g5) \geq 2 \\
 & \text{acceleration } (g2) < 30
 \end{aligned}$$

372 To achieve this in R, the following steps are proposed:

```
373 # filter out cars which do not
374 # respect Thierry's initial rules
375
376 fPT <- pT[(pT$(g4)>=2 & pT$(g5)>=2 & pT$(g2) < 30), ]
```

377 Furthermore, Thierry notices that car a11 (P309-16) is at least as good as car
378 a14 (R21t) on all the criteria, and thus he wishes to remove the latter.

```
379 # drop car a14 from the table
380
381 fPT <- fPT[!(rownames(fPT) %in% "a14"), ]
```

382 The resulting filtered performance table is shown by typing `fPT` on the command
383 prompt:

```
384           g1  g2  g3  g4  g5
385 a03      16973 29.0 34.9 2.66 2.50
386 a07      18971 28.0 35.6 2.33 2.00
387 a11      17537 28.3 34.8 2.33 2.75
388 a12      15980 29.6 35.3 2.33 2.75
```

389 Thierry now asks for a graphical representation of the data. We choose to show
390 him first the performances of the remaining alternatives as radar plots. This allows
391 him to compare their performances in a very synthetic way and to become aware
392 of their conflicting evaluations.

393 To achieve this in R, we first store the preference directions of the criteria
394 ("min" if the criterion has to be minimized, "max" if it has to be maximized) in
395 a vector:

```
396 criteriaMinMax <- c("min", "min", "min", "max", "max")
397
398 names(criteriaMinMax) <- colnames(pT)
```

399 Radar plots can display the preferred values on the outside of the radar and the
400 less preferred values in the center of the graph. We can use the following code to
401 create a radar plot of the alternatives:

```
402 library(MCDA)
403 plotRadarPerformanceTable(fPT, criteriaMinMax,
404                           overlay=FALSE, bw=TRUE, lwd =5)
```

405 The resulting plots (Figure 3) are shown to Thierry. He notices that a12 (P309)
406 is the best car in terms of price and road-hold, but that it has quite bad evaluations
407 for the acceleration, pick-up and brakes criteria. a03 (Sunny) and a11 (P309-16)
408 seem to be much more well-balanced, whereas a07 (Civic) is only good on the
409 acceleration criterion.

410 All in all, Thierry considers that his filtering rules have probably been too
411 strict, and that he wishes to continue the analysis with all the initial alternatives.

412 We continue illustrating the decision aiding process in Figure 4. We have now
413 entered the third stage of the process, that of constructing the evaluation model.
414 We denote with 2.a Thierry's decision to use filtering rules and with 2.b the defi-
415 nition of these rules. The construction of the radar plots are depicted through step
416 2.c, while the decision to not validate the model is given by step 2.d.

417 4.2.2 Weighted sum

418 Thierry now proposes to see how the alternatives compare to each other with
419 respect to each criterion. Among other things, he wishes to determine which al-
420 ternatives have the best and worst evaluations on the criteria.

421 We therefore suggest to plot the values taken by the alternatives in barcharts,
422 for each of the criteria. Such a function is not implemented in the MCDA package

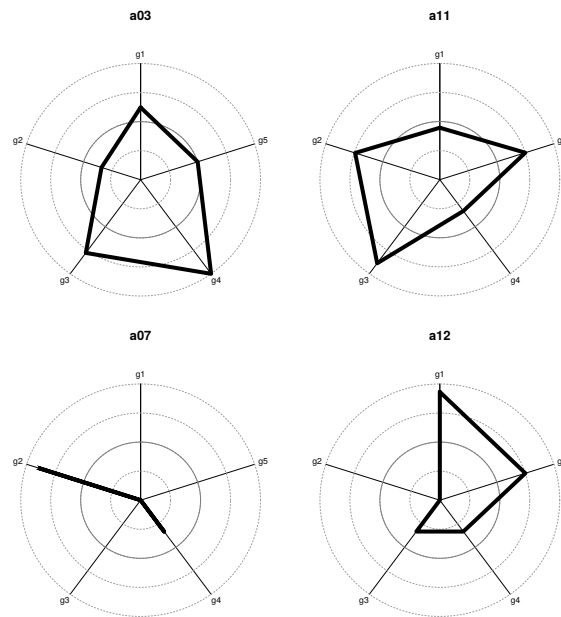


Fig. 3 Radar graphs of the 4 alternatives obtained after the filtering.

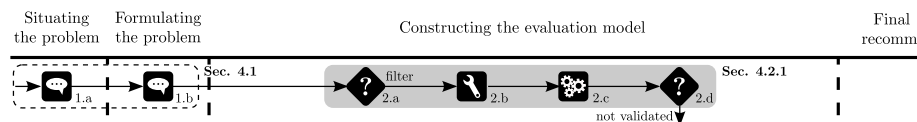


Fig. 4 Second part of Thierry's decision aiding process.

423 because base R provides this functionality already. We therefore use the following
424 code to generate the 5 plots:

```
425 par(mfrow=c(2,3))
426 for (i in 1:dim(pT)[2]){
427   yaxis <- range(pT[,i])*c(0.99,1.05)
428   if (criteriaMinMax[i] == "min")
429     oPT <- pT[order(pT[,i],decreasing=FALSE),]
430   else
431     oPT <- pT[order(pT[,i],decreasing=TRUE),]
432   name <-paste(colnames(pT)[i], "(",criteriaMinMax[i],")", sep="")
433   barplot(oPT[,i], main=name, names.arg = rownames(oPT),
434           density = i*10, ylim = yaxis, xpd=FALSE)
435 }
```

436 Thierry analyzes the resulting plots, shown in Figure 5. The alternatives labeled
437 on the horizontal axis are ordered from left to right according to the preferential
438 direction. He observes, among other things, that alternative a11 (P309-16) seems
439 to be a good alternative, as it performs well on many of criteria (except g_1 (price)).
440 He seems to be very interested in this alternative and suggests that the rather bad
441 performance on the price criterion could be compensated by the good performances
442 on the other criteria.

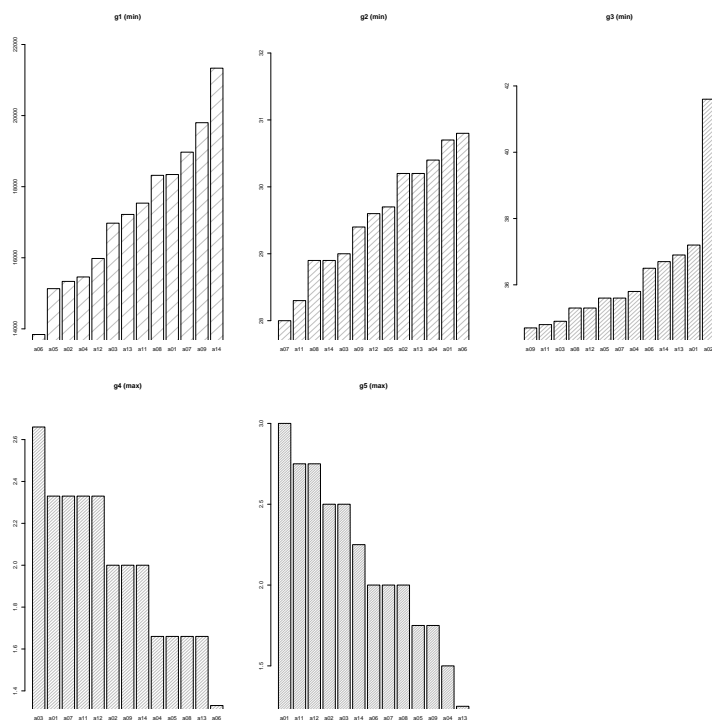


Fig. 5 Bar plots of the performances for each of the 5 criteria.

We deduce from this first discussion with Thierry that he wishes to maximize a quantity which we could call the “value” of the cars. Consequently, our goal in the next steps of the decision aiding process will be to construct a single “super-scale” which reflects the value system of Thierry and his preferences. If we write \succsim for the overall preference relation of Thierry on the set of cars, the goal will be to determine a value function u that allows us to rank the alternatives and represent Thierry’s preferences, i.e., which satisfies

$$a \succsim b \iff u(a) \geq u(b).$$

443 for all alternatives a and b .

444 The value $u(a)$ depends naturally on the evaluations $\{g_i(a), i = 1, \dots, n\}$ of
 445 alternative a (where n is the number of criteria).

446 Thierry suggests to use a weighted sum to aggregate the various evaluations of
 447 the alternatives on the criteria. As described in [Bouyssou et al \(2000\)](#), he chooses
 448 to normalize the data (each criterion at a time) by dividing each evaluation by the
 449 highest value obtained on the corresponding criterion. He then assigns weights to
 450 the criteria according to [Table 2](#). The first three criteria receive negative weights
 451 since they have to be minimized.

452 The above described normalization is done via a function from the MCDA
 453 package for R:

	cost ($g1$, €)	accel. ($g2$, s)	pick up ($g3$, s)	brakes ($g4$)	road-hold ($g5$)
weight	-1	-2	-1	0.5	0.5

Table 2 Thierry's naive weights for the weighted sum model.

car ID	car name	cost ($g1$, €)	accel. ($g2$, s)	pick up ($g3$, s)	brakes ($g4$)	road-holding ($g5$)
a10	R19	16966	30	37.7	2.33	3.25

Table 3 Supplementary car for Thierry's car selection problem.

```

454 # normalization of the data from the performance table
455
456 normalizationTypes <- c("percentageOfMax", "percentageOfMax",
457   "percentageOfMax", "percentageOfMax",
458   "percentageOfMax")
459
460 names(normalizationTypes) <- c("g1", "g2", "g3", "g4", "g5")
461
462 nPT <- normalizePerformanceTable(pT, normalizationTypes)

```

Then, the weighted sum is calculated as follows :

```

464 # weights and the weighted sum
465
466 w <- c(-1, -2, -1, 0.5, 0.5)
467 names(w) <- colnames(pT)
468 ws <- weightedSum(nPT, w)

```

The ranks of the alternatives can be derived from ws by typing:

```

470 # rank the scores of the alternatives
471 rank(-ws)

```

This produces :

```

473 a01 a02 a03 a04 a05 a06 a07 a08 a09 a11 a12 a13 a14
474 5 6 2 10 7 9 4 8 11 1 3 13 12

```

Thierry observes that the best car, according to this aggregation method, is a11, before a03. He however discovers that one potential car has been forgotten in this decision aiding process. It is given in Table 3.

Note that this car is labelled a10, in accordance with the data from [Bouyssou et al \(2000\)](#).

This car is added to the performance table as follows:

```

481 # add supplementary car to pT
482
483 missing <- c(16966, 30, 37.7, 2.33, 3.25)
484 pT <- rbind(pT, missing)
485 rownames(pT)[14] <- "a10"

```

This new performance table is then normalized and a weighted sum is calculated on each alternative:

```

488 # normalization
489

```

```

490 nPT <- normalizePerformanceTable(pT,normalizationTypes)
491
492 # weighted sum
493
494 ws<-weightedSum(nPT,w)

```

The ranking of the alternatives is then shown to Thierry as follows:

```

496 # rank the scores of the alternatives
497 rank(-ws)

```

This produces :

```

499 a01 a02 a03 a04 a05 a06 a07 a08 a09 a11 a12 a13 a14 a10
500 6 8 1 11 7 10 5 9 12 2 3 14 13 4

```

This time, car a03 is considered as the best, before car a11. Thierry is surprised that adding alternative a10 to the performance table produced a rank reversal between the first two alternatives of the ranking. This is due to the normalization method, which depends on the data which is present in the performance table. We recommend Thierry to use a more complex model of his preferences, which is independent of the data of the performance table.

We fill the previously presented steps in the decision aiding process in Figure 6.

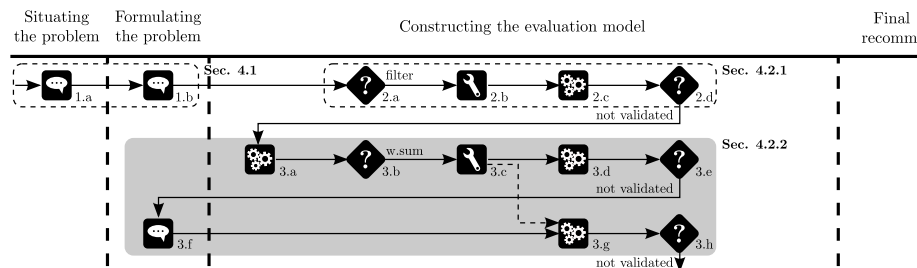


Fig. 6 Third part of Thierry's decision aiding process.

After not validating the previous model, Thierry looks closer at the existing data in 3.a. Based on his remarks, the analyst decides in 3.b to use a weighted sum in order to model his preferences. Thierry gives his relative preferences in 3.c, which are then used in 3.d to compute a ranking of the cars. Before validating this model in 3.e, Thierry realizes that he forgot to include a car in his decision. This takes us back to the second stage of the decision aiding process, as we are identifying other alternatives that need to be included in the model (step 3.f). We then return to the third stage and use the previously constructed model to generate a new ranking of the alternatives in step 3.g. Thierry notices a rank reversal, which prompts him to not validate this model in step 3.h.

4.2.3 MAVT

We choose to construct a model of Thierry's preferences through an additive model, aggregating some marginal value functions on the original evaluations via a weighted sum (the weights representing trade-offs between the criteria).

522 Now that a motivated choice has been made on the preference model, the next
 523 step of this decision aiding process is to elicit the preferences of Thierry (with
 524 respect to this additive value model). To determine the marginal value functions,
 525 a direct method could be used (by direct numerical estimations, or by indifference
 526 judgements). However, as he seems to be quite an expert in sports cars, we decide
 527 to switch to an indirect elicitation method, where the shapes of the marginal value
 528 functions and the trade-offs are inferred from Thierry's overall preferences on some
 529 cars.

530 The chosen disaggregation method is UTA and was described by [Jacquet-](#)
 531 [Lagrèze and Siskos \(1982\)](#). It searches for piecewise linear marginal value functions
 532 which respect the input preferences expressed by the decision maker. In our case,
 533 these a priori preferences are represented by a preorder on a subset of cars, that
 534 Thierry knows quite well (the learning set). Thierry chooses to rank 5 cars as
 535 follows:

$$a_{11} \succ a_{03} \succ a_{13} \succ a_{09} \succ a_{14}.$$

536 In the MCDA package for R, the UTA disaggregation method can be called
 537 through the UTA function. Its arguments are the performance table, the preference
 538 directions for each criterion, the number of breakpoints for the piecewise linear
 539 value functions, a separation threshold (representing the minimal difference in
 540 value between two consecutive alternatives from the learning set) and the lower
 541 and upper bounds of the criteria scales. For arguments of parsimony, we decide to
 542 search for piecewise linear value functions with 1 segment.

```
543 # ranks of the alternatives
544 alternativesRanks <- c(1,2,3,4,5)
545 names(alternativesRanks) <- c("a11","a03","a13","a09","a14")
546
547 # number of break points for each criterion : 1 segment = 2 breakpoints
548 criteriaNumberOfBreakPoints <- c(2,2,2,2,2)
549 names(criteriaNumberOfBreakPoints) <- colnames(pT)
550
551 # lower bounds of the criteria for the determination of value functions
552 criteriaLBs=apply(pT,2,min)
553 names(criteriaLBs) <- colnames(pT)
554
555 # upper bounds of the criteria for the determination of value functions
556 criteriaUBs=apply(pT,2,max)
557 names(criteriaUBs) <- colnames(pT)
558
559 # the separation threshold
560 epsilon <- 0.01
561
562 x<-UTA(pT, criteriaMinMax,
563       criteriaNumberOfBreakPoints, epsilon,
564       alternativesRanks = alternativesRanks,
565       criteriaLBs = criteriaLBs, criteriaUBs = criteriaUBs)
```

571 The calculation is successful, and the result is shown by typing x on the com-
 572 mand prompt:

```

573 $optimum
574 [1] 0
575
576 $valueFunctions
577 $valueFunctions$g1
578   [,1] [,2]
579 x 21334 1.38410e+04
580 y    0 4.61114e-01
581
582 $valueFunctions$g2
583   [,1] [,2]
584 x 30.8 28
585 y 0.0 0
586
587 $valueFunctions$g3
588   [,1] [,2]
589 x 41.6 34.7000000
590 y 0.0 0.2049873
591
592 $valueFunctions$g4
593   [,1] [,2]
594 x 1.33 2.66
595 y 0.00 0.00
596
597 $valueFunctions$g5
598   [,1] [,2]
599 x 1.25 3.2500000
600 y 0.00 0.3338987
601
602
603 $overallValues
604   a03 a09 a11 a13 a14
605 0.67611 0.38286 0.68611 0.39286 0.31252
606
607 $ranks
608 a03 a09 a11 a13 a14
609 2 4 1 3 5
610
611 $errors
612 a03 a09 a11 a13 a14
613 0 0 0 0 0
614
615 $Kendall
616 [1] 1

```

617 The structure returned by the UTA function is a list / dictionary containing
618 the following main elements:

- 619 – optimum : the value of the objective function in the UTA algorithm;
- 620 – valueFunctions : a list containing the value function for each criterion;
- 621 – overallValues : the overall values of the learning set;
- 622 – ranks : the ranks of the elements of the learning set;
- 623 – error : the errors which have to be added to the overall values of the alter-
624 natives of the learning set in order to respect the input order;
- 625 – Kendall : Kendall's rank correlation index between the input and the output
626 ranking of the learning set.

627 We can observe that Thierry's ranking is compatible with the chosen model
628 (Kendall's rank correlation index equals 1, there are no errors, and the optimal

629 value of the objective function equals 0). We plot the obtained value functions as
 630 follows:

```
631 # plot the piecewise linear value functions
632
633 plotPiecewiseLinearValueFunctions(x$valueFunctions)
```

634 The resulting marginal value functions are shown on Figure 7. The maximal
 635 value on the ordinate axis represents the trade-off weight in the aggregation.

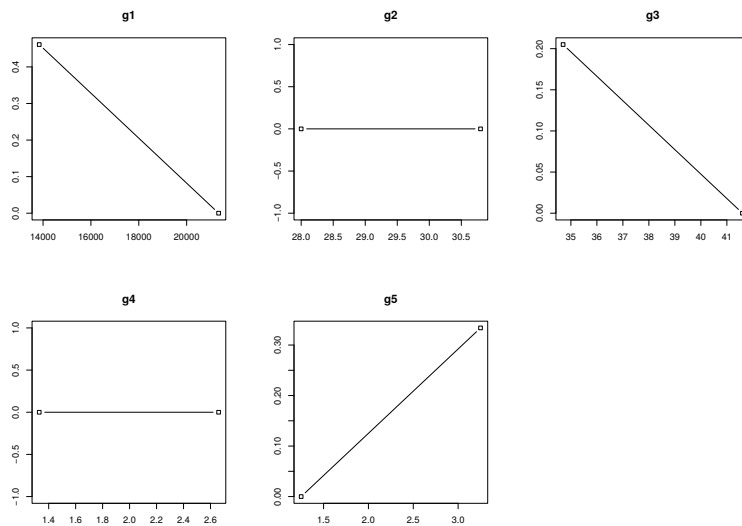


Fig. 7 Marginal value functions for the criteria with respect to the initial reference ranking.

636 Thierry is not totally convinced by this preference model. He agrees that the
 637 price is very important in the aggregation, but he considers that the accelera-
 638 tion should also be considered to discriminate between alternatives. He decides to
 639 modify his a priori ranking by adding two alternatives:

$$a11 \succ a03 \succ a08 \succ a04 \succ a13 \succ a09 \succ a14.$$

640 The following lines of code are entered in R :

```
641 # ranks of the alternatives for the second try
642
643 alternativesRanks <- c(1,2,3,4,5,6,7)
644 names(alternativesRanks) <- c("a11", "a03", "a08", "a04", "a13", "a09", "a14")
645
646 x2<-UTA(pT, criteriaMinMax,
647         criteriaNumberOfBreakPoints, epsilon,
648         alternativesRanks = alternativesRanks,
649         criteriaLBs = criteriaLBs, criteriaUBs = criteriaUBs)
650
651 # plot the piecewise linear value functions
652
653 plotPiecewiseLinearValueFunctions(x2$valueFunctions)
```

654 The new calculations generate the value functions represented in Figure 8. This
 655 time Thierry validates the model, as the acceleration criterion plays a significant
 656 role in the aggregation.

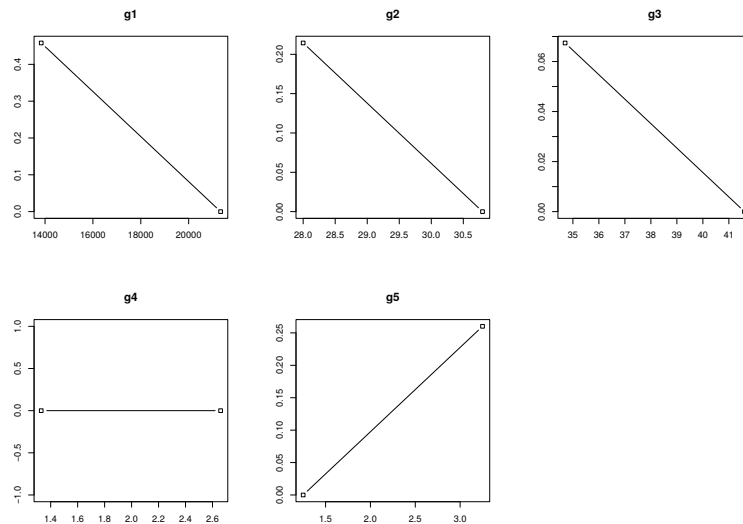


Fig. 8 Marginal value functions for the criteria after the update of the a priori ranking of Thierry.

657 Now that a model of Thierry’s preferences has been found, these marginal
 658 value functions can be used to rank all the cars. This is done by applying the
 659 value functions on the original performance table, and by performing an additive
 660 aggregation of the marginal values vector, for each alternative. In the MCDA
 661 package for R, this is done as follows:

```
662 # apply the value functions on the original performance table
663
664 tPT <- applyPiecewiseLinearValueFunctionsOnPerformanceTable(
665     x2$valueFunctions,
666     pT
667 )
668
669 # calculate the overall score of each alternative
670
671 mavt <- weightedSum(tPT, rep(1, 5))
```

672 The second argument of the `weightedSum` function is a vector of equi-important
 673 weights, as the trade-off weight is already contained in the value functions. The
 674 output of the `weightedSum` function is the “super-scale” we were mentioning
 675 earlier (page 13). It indicates, provided it can be considered as accurate, the value
 676 of each car, according to Thierry’s preference model.

677 These overall scores can be obtained by typing `mavt` in the command prompt:

```
678   a01   a02   a03   a04   a05   a06   a07
679 0.4611504 0.5752482 0.6324617 0.4788993 0.5870830 0.6054313 0.5150286
680   a08   a09   a10   a11   a12   a13   a14
```

681 0.4888993 0.3334222 0.6265008 0.6850774 0.6758266 0.3434222 0.3234222

682 We can observe that the car which obtains the highest score is a11 (P309-16).
 683 This confirms Thierry’s preliminary analysis.

684 Note here that after the confrontation of the decision maker to the overall
 685 scores, one could easily imagine a scenario where Thierry is not satisfied with the
 686 result, and that he wishes to update the preference model. To avoid adding com-
 687 plexity to this fictive decision aiding process, we suppose that Thierry is satisfied
 688 with the scores.

689 A further step of the decision aiding process is to analyze the result, and to
 690 plot some graphical summaries of the outputs. In a more complex process, this
 691 phase could also be completed by a sensitivity or robustness analysis. It could
 692 also be the right place to compare the outputs of various aggregation models (for
 693 example, the ELECTRE methods, see Bouyssou et al (2000), Chapter 6, or Meyer
 694 and Bigaret (2012a) for the PROMETHEE methods).

695 Here, we mainly confront Thierry with the ranking of the cars according to
 696 their overall scores,

697 We complete the R code by calling a function to plot the ranking of the cars:

698 `plotAlternativesValuesPreorder(mavt, decreasing=TRUE)`

699 Figure 9 shows the first 7 positions of this ranking.

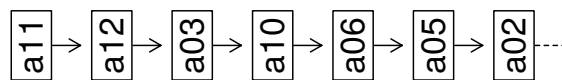


Fig. 9 The ranking obtained by the additive value model.

700 According to this model, car a11 is ranked first, before car a03 and a12.

701 We finalize the depiction of the decision aiding process of this illustrative ex-
 702 ample in Figure 10.

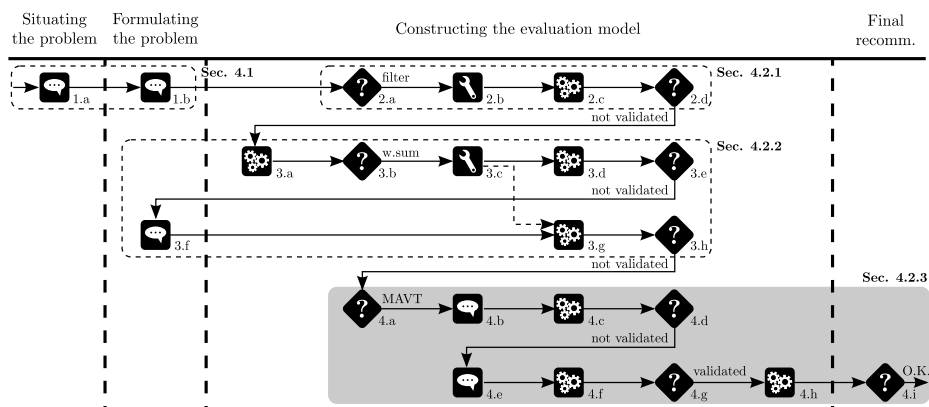


Fig. 10 Last part of Thierry’s decision aiding process.

703 Following the decision to use another preference model, the choice of MAVT
704 is given in step 4.a. The preference elicitation step is depicted in step 4.b, while
705 the application of UTA to generate the marginal value functions is depicted in
706 step 4.c. The illustration of these functions and the subsequent decision of Thierry
707 to not validate this result is given in 4.d. In 4.e we update the partial ranking
708 given by Thierry, in 4.f we generate the updated marginal value functions, while
709 in 4.g Thierry validates the model. We then continue with generating the final
710 ranking in step 4.h, using this model, while in 4.i we step into the final recommen-
711 dation phase, where Thierry is confronted with this ranking. Thierry validates the
712 recommendation and therefore the process is finished.

713 5 Discussion and conclusion

714 In this paper, we proposed to support the MCDA process throughout all of
715 its steps by use of a single environment, the R statistical software. Currently,
716 analysts and the DMs have to resort to using multiple tools at different stages of
717 the decision aiding process, moving from one to the other, adding an additional
718 level of difficulty. The choice of using R throughout the process was motivated
719 by its focus towards data analysis, its open-source and package-based philosophy,
720 as well as its large community of users and contributors. Furthermore, we have
721 developed the MCDA package which seeks to encompass as many of the MCDA
722 algorithms as possible in order to provide additional support. We have illustrated
723 the use of R and the MCDA package using a well-known illustrative example from
724 the literature and in addition highlighting the different steps that were undertaken
725 within the MCDA process. We have shown that, even when the process is complex,
726 by using R and the MCDA package we are able to successfully achieve a solution.

727 While currently, the MCDA package contains algorithms linked to only a few
728 methods, we wish to continue developing it in the future so that as many of
729 the MCDA algorithms can be found within it. Furthermore, functions linked to
730 the presentation of the results, for instance graphically, will also be added to
731 complement the existing ones. We additionally wish to continue applying this
732 methodology and these tools to other practical applications.

733 References

- 734 Baizyldayeva U, Vlasov O, Kuandykov AA, Akhmetov TB, et al (2013) Multi-
735 criteria decision support systems: Comparative analysis. *Middle-East Journal*
736 *of Scientific Research* 16(12):1725–1730
- 737 Belton V, Stewart T (2002) *Multiple criteria decision analysis: an integrated ap-
738 proach*. Springer Science & Business Media
- 739 Bisdorff R, Dias LC, Meyer P, Pirlot M, Mousseau V (2015) *Evaluation and Deci-
740 sion Models With Multiple Criteria: Case Studies*. International Handbooks on
741 Information Systems, Springer, Berlin
- 742 Bogetoft P, Otto L (2015) *Benchmarking-Benchmark and Frontier Analysis Using
743 DEA and SFA*. <https://cran.r-project.org/package=Benchmarking>
- 744 Bouyssou D, Marchant T, Pirlot M, Perny P, Tsoukiàs A, Vincke P (2000) *Eval-
745 uation and decision models: A critical perspective*. Kluwer, Dordrecht

- 746 Bouyssou D, Marchant T, Pirlot M, Tsoukiàs A, Vincke P (2006) Evaluation and
747 decision models with multiple criteria: Stepping stones for the analyst, 1st edn.
748 International Series in Operations Research and Management Science, Volume
749 86, Boston
- 750 Clemen RT, Reilly T (2001) Making hard decision with decision tools. South-
751 Western Cengage Learning, Mason, Ohio
- 752 Coutinho-Rodrigues J, Simão A, Antunes CH (2011) A gis-based multicriteria
753 spatial decision support system for planning urban infrastructures. *Decision*
754 *Support Systems* 51(3):720–726
- 755 Dias LC, Mousseau V (2003) Iris: A dss for multiple criteria sorting problems.
756 *Journal of Multi-Criteria Decision Analysis* 12(4-5):285–298
- 757 Figueira J, Greco S, Ehrgott M (2005) Multiple criteria decision analysis: state of
758 the art surveys, vol 78. Springer Science & Business Media
- 759 Gentry J, Long L, Gentleman R, Falcon S, Hahne F, Sarkar D, Rgraphviz KH
760 (2009) Provides plotting capabilities for r graph objects. R package version 2(0)
- 761 Gentry J, Gentleman R, Huber W (2016) How to plot a graph using
762 rgraphviz. [http://www.bioconductor.org/packages/release/bioc/vignettes/R-](http://www.bioconductor.org/packages/release/bioc/vignettes/Rgraphviz/inst/doc/Rgraphviz.pdf)
763 [graphviz/inst/doc/Rgraphviz.pdf](http://www.bioconductor.org/packages/release/bioc/vignettes/Rgraphviz/inst/doc/Rgraphviz.pdf)
- 764 Grabisch M, Kojadinovic I, Meyer P (2006) Using the Kappalab R package for
765 capacity identification in choquet integral based maut. In: Proceedings of the
766 11th international conference on information processing and management of
767 uncertainty in knowledge-based systems, pp 1702–1709
- 768 Grabisch M, Kojadinovic I, Meyer P (2015) kappalab - Non-Additive Measure and
769 Integral Manipulation Functions. <https://cran.r-project.org/package=kappalab>
- 770 Guitouni A, Martel JM, Vincke P, North P, Val-bblair O (1998) A framework to
771 choose a discrete multicriterion aggregation procedure. Defence Research Estab-
772 lishment Valcatier (DREV)
- 773 Hodgett RE (2016) Comparison of multi-criteria decision-making methods for
774 equipment selection. *The International Journal of Advanced Manufacturing*
775 *Technology* 85(5):1145–1157, DOI 10.1007/s00170-015-7993-2
- 776 Hodgett RE, Martin EB, Montague G, Talford M (2014) Handling uncertain deci-
777 sions in whole process design. *Production Planning & Control* 25(12):1028–1038,
778 DOI 10.1080/09537287.2013.798706
- 779 Hwang CL, Yoon K (1981) Multiple Attribute Decision Making: Methods and Ap-
780 plications A State-of-the-Art Survey (Lecture Notes in Economics and Mathe-
781 matical Systems). Springer
- 782 IEEE Spectrum (2016) The 2016 top programming languages, available from:
783 [http://spectrum.ieee.org/computing/software/the-2016-top-programming-](http://spectrum.ieee.org/computing/software/the-2016-top-programming-languages)
784 [languages](http://spectrum.ieee.org/computing/software/the-2016-top-programming-languages)
- 785 Ihaka R, Gentleman R (1996) R: a language for data analysis and graphics. *Journal*
786 *of computational and graphical statistics* 5(3):299–314
- 787 International Society on Multiple Criteria Decision Making (2014)
788 Multiple criteria decision making website, available from:
789 <http://www.mcdmsociety.org/content/software-related-mcdm>
- 790 Ishizaka A, Nemery P (2013) Multi-method platforms. *Multi-Criteria Decision*
791 *Analysis: Methods and Software* pp 275–287
- 792 Jacquet-Lagrèze E, Siskos Y (1982) Assessing a set of additive utility functions
793 for multicriteria decision making: the UTA method. *European Journal of Oper-*
794 *ational Research* 10:151–164

- 795 Keeney R, Raiffa H (1976) Decisions with multiple objectives: Preferences and
796 value tradeoffs. J. Wiley, New York
- 797 Kostkowski M, Slowinski R (1996) UTA+ application (v. 1.20)-user's manual.
798 Document du LAMSADE 95
- 799 Lahdelma R, Salminen P, Hokkanen J (2014) Using Multicriteria Methods in Envi-
800 ronmental Planning and Management. *Environmental Management* 26(6):595–
801 605, DOI 10.1007/s002670010118
- 802 Leistedt B (2011) UTAR library for MCDA. [https://cran.r-](https://cran.r-project.org/package=UTAR)
803 [project.org/package=UTAR](https://cran.r-project.org/package=UTAR)
- 804 Leroy A, Mousseau V, Pirlot M (2011) Learning the parameters of a multiple
805 criteria sorting method. In: Brafman RI, Roberts FS, Tsoukiàs A (eds) ADT,
806 Springer, Lecture Notes in Computer Science, vol 6992, pp 219–233
- 807 Make It Rational (2016) Make it rational website, available from: makeitrat-
808 [tional.com/](http://makeitrat-tional.com/)
- 809 Mayag B, Cailloux O, Mousseau V (2011) Mcda tools and risk analysis: the decision
810 deck project. *Advances in Safety, Reliability and Risk Management: ESREL*
811 2011 p 377
- 812 Meyer P, Bigaret S (2012a) diviz: a software for modeling, processing and sharing
813 algorithmic workflows in MCDA. *Intelligent Decision Technologies* 6(4):283–296,
814 doi:10.3233/IDT-2012-0144
- 815 Meyer P, Bigaret S (2012b) RXMCDA - Functions to Parse and Create XMCDA
816 Files. <https://cran.r-project.org/package=RXMCDA>
- 817 Meyer P, Olteanu AL (2017) Integrating large positive and negative performance
818 differences into multicriteria majority-rule sorting models. *Computers & Oper-*
819 *ations Research* pp 216 – 230
- 820 Meyer P, Bigaret S, Hodgett R, Olteanu AL (2015) MCDA-Functions
821 to Support the Multicriteria Decision Aiding Process. [https://cran.r-](https://cran.r-project.org/package=MCDA)
822 [project.org/package=MCDA](https://cran.r-project.org/package=MCDA)
- 823 Mousseau V, Slowinski R, Zielniewicz P (1999) ELECTRE TRI 2.0 a methodolog-
824 ical guide and user's manual. Document du LAMSADE 111:263–275
- 825 Mousseau V, Slowinski R, Zielniewicz P (2000) A user-oriented implementation
826 of the ELECTRE-TRI method integrating preference elicitation support. *Com-*
827 *puters & Operations research* 27(7):757–777
- 828 Mustajoki J, Marttunen M (2013) Comparison of multi-criteria decision analytical
829 software. Finnish Environment Institute
- 830 Papamichail KN, French S (2013) 25 years of MCDA in nuclear emergency man-
831 agement. *IMA Journal of Management Mathematics* 24(4):481 – 503
- 832 Piatetsky G (2016) R, Python duel as top analytics, data science
833 software – kdnuggets 2016 software poll results, available from:
834 <http://www.kdnuggets.com/2016/06/r-python-top-analytics-data-mining->
835 [data-science-software.html](http://www.kdnuggets.com/2016/06/r-python-top-analytics-data-mining-data-science-software.html)
- 836 R Development Core Team (2008) R: A Language and Environment for Statistical
837 Computing. R Foundation for Statistical Computing, Vienna, Austria, URL
838 <http://www.R-project.org>, ISBN 3-900051-07-0
- 839 Roy B (1991) The outranking approach and the foundations of electre methods.
840 *Theory and Decision* 31(1):49–73, DOI 10.1007/BF00134132
- 841 Roy B (1996) *Multicriteria Methodology for Decision Aiding*. Kluwer Academic,
842 Dordrecht

- 843 Saaty TL (1980) *The Analytic Hierarchy Process: Planning, Priority Setting, Re-*
844 *source Allocation (Decision Making Series)*. McGraw-Hill (Tx)
- 845 Simon HA (1976) *Administrative behavior; a study of decision-making processes*
846 *in administrative organization-3*
- 847 Siraj S, Mikhailov L, Keane JA (2015) Contribution of individual judgments to-
848 ward inconsistency in pairwise comparisons. *European Journal of Operational*
849 *Research* 242(2):557–567, DOI 10.1016/j.ejor.2014.10.024
- 850 Sobrie O, Mousseau V, Pirlot M (2013) Learning a majority rule model from large
851 sets of assignment examples. In: ADT, Springer, *Lecture Notes in Computer*
852 *Science*, vol 8176, pp 336–350
- 853 Statistical Design Institute (2016) Topsis website, available from:
854 <http://www.stat-design.com/Software/TOPSIS.html>
- 855 Taillandier P, Stinckwich S (2011) Using the promethee multi-criteria decision
856 making method to define new exploration strategies for rescue robots. In: 2011
857 IEEE International Symposium on Safety, Security, and Rescue Robotics, pp
858 321–326, DOI 10.1109/SSRR.2011.6106747
- 859 Tervonen T (2012) JSMAA: open source software for smaa computations. *Int J*
860 *Syst Sci* 2012:1–13
- 861 TransparentChoice Ltd (2016) Transparent choice website, available from:
862 <https://www.transparentchoice.com>
- 863 Tsoukias A (2007) On the concept of decision aiding process: an operational per-
864 spective. *Annals of Operations Research* 154:3 – 27
- 865 Tversky A, Kahneman D (1981) The framing of decisions and the psychology of
866 choice. *Science* 211(4481):453–458
- 867 Venables B, Smith D, Gentleman R, Ihaka R (1998) *Notes on R: a programming*
868 *environment for data analysis and graphics*. University of Auckland
- 869 von Winterfeldt D, Edwards W (1986) *Decision Analysis and Behavioral Research*.
870 Cambridge University Press, Cambridge
- 871 Wahlster P, Goetghebuer M, Kriza C, Niederländer C, Kolominsky-Rabas P (2015)
872 Balancing costs and benefits at different stages of medical innovation: a system-
873 atic review of multi-criteria decision analysis (MCDA). *BMC Health Services*
874 *Research* 15(1):1 – 12
- 875 Weistroffer HR, Smith CH, Narula SC (2005) Multiple criteria decision support
876 software. In: *Multiple criteria decision analysis: state of the art surveys*, Springer,
877 pp 989–1009
- 878 Yatsalo B, Didenko V, Gritsyuk S, Sullivan T (2015) Decerns: A framework for
879 multi-criteria decision analysis. *International Journal of Computational Intelli-*
880 *gence Systems* 8(3):467–489