On the elicitation of criteria weights in PROMETHEE-based ranking methods for a mobile application

Francesco Lolli[1], Elia Balugani*, Alessio Ishizaka**, Rita Gamberini*, Maria Angela Butturi*, Samuele Marinello*, Bianca Rimini**

Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Via Amendola 2 – Padiglione Morselli, 42100 Reggio Emilia, Italy

*** Centre of Operations Research and Logistics, Portsmouth Business School, University of Portsmouth, Richmond Building, Portland Street, Portsmouth PO1 3DE, United Kingdom*

francesco.lolli@unimore.it; *elia.balugani@unimore.it*; *alessio.ishizaka@port.ac.uk*; *rita.gamberini@unimore.it*; *mariaangela.butturi@unimore.it*; *samuele.marinello@unimore.it*; *bianca.rimini@unimore.it*

Abstract

Today, almost everybody has a smartphone and applications have been developed to help users to take decisions (e.g. which hotel to choose, which museum to visit, etc). In order to improve the recommendations of the mobile application, it is crucial to elicit the preference structures of the user. As problems are often based on several criteria, multicriteria decision aiding methods are most adequate in these cases, and past works have proposed indirect eliciting approaches for multicriteria decision aiding methods. However, they often do not aim of reducing as much as possible the cognitive efforts required by the user. This is prerequisite of mobile applications as they are used by everybody. In this work, the weights to assign to the evaluation criteria in a PROMETHEE-based ranking approach are unknown, and therefore must be elicited indirectly either from a partial ranking provided by the user or from the selection of his/her most preferred alternative into a subset of reference alternatives. In the latter case, the cognitive effort required by the decision-maker is minimal. Starting from a linear optimisation model aimed at searching for the most discriminating

[1] Corresponding author. Tel.: +39 0522522635
E-mail address: francesco.lolli@unimore.it

vector of weights, three quadratic variants are proposed subsequently to overcome the issues arising from the linear model. An iterative quadratic optimisation model is proposed to fit the real setting in which the application should operate, where the eliciting procedure must be launched iteratively and converge over time to the vector of weights, which are the weights that the user implicitly assigns to the evaluation criteria. Finally, three experiments are performed to confirm the effectiveness and the differences between the proposed models.

Keywords: Multicriteria decision making; Elicitation; Criteria weights; PROMETHEE; outranking models; mobile applications

1. Introduction

Intercepting the needs of Internet users constitutes a challenging goal for many mobile applications. A high response speed is an inescapable requisite if applications are to be desirable to users, but the quality level of the provided service has to be as high as possible to satisfy user needs. The probability of satisfying such needs quickly is thus closely related to the ability to elicit them both efficiently and effectively.

Elicitation involves learning the needs of users, and may be attained through different techniques (e.g., interviews, brainstorming, prototyping and observation) depending on the specific application field (e.g., Hickey and Davis, 2003; Barton, 2015). When the goal of the application is a multicriteria decision, assessing the weights to assign to the criteria is crucial (Riabacke, Danielson, & Ekenberg, 2012). Indeed, the outcomes of multicriteria selection, ranking, and sorting methods are sensitive to the weights assigned to the evaluation criteria within certain stability intervals (Mareschal, 1988). In weighted-sum aggregation models (e.g., MAUT) and outranking approaches (such as ELECTRE and PROMETHEE), the global scores of the alternatives under analysis are achieved by means of the weighted sums of the local scores that the alternatives have on each single criterion, with the weights representing the importance given by the decision-maker (DM) to the criteria. These weights depend on the needs and the preferences of the DM and may be elicited either directly by asking him/her to provide judgments about criteria, or indirectly by using only his/her past decisions.

To increase the usability of mobile applications designed for multicriteria problems, indirect elicitation procedures are typically preferred, because less information and cognitive effort are required from the user. The terms ‘user’ and ‘decision-maker’ are interchangeable in this context. In a multicriteria selection problem that underlies the functioning of a mobile application, a general iterative scheme for an indirect weights elicitation procedure can consist of the following stages:

i) Based on the vector of the criteria weights achieved in the last iteration, a subset of the high-ranking alternatives, named reference alternatives, is provided to the user;

ii) The user either selects the most preferred within this subset or establishes their ranking in terms of holistic preferences among the reference alternatives;

iii) From the information provided at stage ii, the weights elicitation procedure is launched;

iv) A new set of available alternatives is generated when the user accesses the application and a new multicriteria selection problem is presented to the user;

v) The vector of weights obtained at stage iii is used for generating the complete ranking of the current alternatives;

vi) Back to stage i.

The goal is therefore to elicit the criteria weights iteratively from the past decisions of the user.

This general framework requires that the user either selects the most preferred reference alternative at each iteration in the application design, or establishes the ranking of the subset of reference alternatives. In the former case, the most preferred alternative requires the least information from the user, thus conforming to the goal of the application itself. In the latter case, a little more information is required from the user depending on the subset dimension, but again aims to increase the usability of the application as much as possible. The final goal is to converge iteratively to the ‘real’ criteria weight by including added information at each iteration in the form of holistic preferences between reference alternatives. Such a scheme may be framed in the disaggregation paradigm, which is the inductive learning mechanism used in supervised machine learning approaches.

In this work, different optimisation approaches are introduced incrementally, with the aim of reaching an indirect and iterative weights-eliciting approach for PROMETHEE-based ranking models. This approach only makes use of the past decisions of the user (either the most preferred alternative or the ranking of reference alternatives at each iteration), thus reducing as much as possible the cognitive effort required.

Our method was programmed in a mobile application designed during the LUME PlannER project (http://www.lumeplanner.it/index.php/en/), which supports the Emilia Romagna region tourism system. The tourism industry has been revolutionized by the widespread use of smartphones, which has changed the way people search and access information. It enables users to be always and everywhere online and can be used while travelling, supporting the transformation of travel practices that has led to a high degree of unplanned and opportunistic travel behavior (Dickinson et al., 2014).

The developed application allows travellers to plan and implement sustainable tourism routes tailored to their interests. The project focuses on the sustainable goals of promoting widespread regional natural and cultural heritage; managing visitor pressure at main attractions; facilitating mobility and transport by encouraging inter-modality and sustainable transport solutions; enabling accessibility to all itineraries, sites and means of transport; and supporting the development of local commercial and entrepreneurial tourism infrastructure that consists of many micro, small- and medium-sized businesses.

In this study, touristic tours were selected according to multiple criteria (e.g., cost, environmental sustainability, social sustainability, waiting time and accessibility). However, the method is generalizable and can easily be applied to other mobile applications.

This paper is organized as follows: in Section 2 we frame our proposal in terms of the related literature. In Section 3 we outline the classical PROMETHEE method. Section 4 introduces four new variants of eliciting models (LM, QM1, QM2 and DDQM2), with DDQM2 specifically designed to be applied iteratively using the past decisions of the user. In Section 5 the experiments conducted to prove the proposed eliciting methods properties are discussed. The experimental results obtained are presented in Section 6, and Section 7 concludes the paper with a summary of the key findings and suggestions for a further research agenda.

2. Literature review

Criteria weights are crucial in selection, ranking and sorting problems, and several techniques for assigning the weights of evaluation criteria appear in the literature of multicriteria decision making. In particular, surrogate weighting procedures requiring only ordinal information about the criteria priorities (i.e., rank-based approaches) should support the DM when he/she is not confident about translating this information into exact weight values. The different rank-based weighting procedures combined with the PROMETHEE and TOPSIS methods have been compared in the studies of de Almeida Filho, Clemente, Morais and de Almeida (2018) and Alemi-Ardakani, Milani, Yannacopoulos and Shokouhi (2016). In order to determine the criteria weights in ELECTRE type methods Figueira & Roy (2002) revised Simos' approach whose procedure adopts a simple set of cards (Simos, 1990). Conversely, through a posterior analysis, Kaliszewski and Podkopaev (2016) started from a multicriteria ranking of alternatives to derive the criteria weights yielding the same ranking by means of the most commonly used and transparent Simple Additive Weighting (SAW) method, which takes the form of a metamodel.

However, the focus of our proposal is on the weights elicitation when given only the selection of the most preferred alternative, which draws on the research into indirect elicitation.

A pioneering indirect elicitation approach is UTilité Additive (UTA) (Jacquet-Lagreze & Siskos, 1982), in which a linear programming approach is taken that uses indirect preference information provided by the DM in the form of holistic preference and indifferent relations between reference alternatives. UTA was originally designed by assuming the axiomatic basis underlying the Multi Attribute Utility Theory (MAUT), that is, the existence of an additive utility function used to infer into the set of those compatible with the DM's preference. UTA considers piecewise linear marginal value functions on single criteria. UTA^{GMS} (Greco, Mousseau, & Słowiński, 2008) generalises UTA via Robust Ordinal Regression (ROR) by taking into account all additive value functions compatible with the indirect preference information, and by allowing general non-decreasing marginal value functions. In case of interacting criteria, a non-additive ROR with utility evaluated in terms of the Choquet integral was formulated by Angilella, Greco and Matarazzo (2010). Here, the DM must provide holistic pairwise preference comparisons of reference alternatives, the intensity of preference on pairs of alternatives, pairwise comparisons on the importance of criteria and the sign and intensity of interaction among criteria. The extension of UTA^{GMS} to sorting problems is the $UTADIS^{GMS}$ method (Greco, Mousseau, & Słowiński, 2010), in which reference alternatives are assigned by the DM to one or several contiguous classes, and then used to build a preference model of the DM represented by a set of general added value functions compatible with the assignment references. The selection of one value function representing all compatible value functions has been addressed by Greco, Kadziński and Słowiński (2011) who used an interactive procedure under the $UTADIS^{GMS}$ approach; the DM is again allowed to assign reference alternatives to ordered classes and to choose the targets she/he would like to account for.

ROR for indirect elicitation is also widely used in outranking methods, as the cognitive effort required by the DM is reduced. The first implementation of ROR to outranking methods places

ELECTRE in the context of choice and ranking problems and is named ELECTRE^{GKMS} (Greco, Kadziński, Mousseau, & Słowiński, 2011), where the DM provides pairwise comparisons stating the truth or falsity of the outranking relation for reference alternatives, along with intervals of indifference and preference thresholds. A different elicitation procedure via optimisation for ELECTRE TRI, i.e., the sorting ELECTRE-based approach, has been formulated by Mousseau and Slowinski (1998), where the holistic judgements provided by the DM refer to the assignment of reference alternatives to classes. The goal is again to infer the parameters of the preference model. Similarly, Dias and Mousseau (2005) limited their approach to an ELECTRE-based interactive process of the partial inference problem, focusing on the inference of discordance-related parameters. ROR has been also applied to PROMETHEE: Kadziński, Greco and Słowiński (2012) introduced PROMETHEE^{GKS}, which uses ROR to identify a set of compatible outranking models, each of which leads to different complete rankings. The best and the worst ranks shown by an alternative on the basis of the compatible outranking models are then achieved by means of mixed integer linear programs, whereas linear programs are enough to determine its extreme net flows. Valuable information on the robustness of the ranks occupied by the alternatives is thus provided to the DM. As PROMETHEE-based outranking methods are based on preference functions that depend on preference and indifference thresholds on each criterion, the DM is assumed to be able to provide the values in the form of intervals. ROR was also extended by Corrente, Greco and SŁowiński (2013) to ELECTRE- and PROMETHEE-based methods in the case of the hierarchy of criteria.

In addition to ROR, Stochastic Multiobjective Acceptability Analysis (SMAA) (Lahdelma, Hokkanen, & Salminen, 1998) explores a large number of parameters compatible with the preference information that the DM implicitly provides, and thus is categorised as an indirect elicitation approach. SMAA considers the probability distributions over the space of all compatible weights and evaluations of alternatives when the DM is able to provide different partial or complete information. Several applications can be found in the literature (Greco, Ishizaka, Matarazzo, &

Torrisi, 2018; De Matteis, Ishizaka, & Resce, 2018). SMAA has been applied to outranking methods, and in particular to ELECTRE both for ranking (Hokkanen, Lahdelma, Miettinen, & Salminen, 1998) and for sorting problems (Tervonen, Figueira, Lahdelma, Dias, & Salminen, 2009), and to PROMETHEE (Corrente, Figueira, & Greco, 2014). The analytic relationship between ROR and SMAA was examined by Kadziński and Tervonen (2013a, 2013b), while Angilella *et al.* (2016) extended ROR and SMAA methods to the case of interacting criteria into a hierarchical structure by recurring to the Choquet integral. To reach an unambiguous ranking of alternatives from the SMAA, Vetschera (2017) developed several models that make use of the SMAA indices.

A further approach specifically focusing on the elicitation of criteria weights within a multicriteria additive model was proposed by De Almeida *et al.* (2016) and termed the Flexible and Interactive Tradeoff (FITradeoff), which overcomes some weaknesses of the traditional trade-off procedure. This traditional procedure and a criticism of it can be found in the studies of Keeney and Raiffa (1993) and Weber and Borcherding (1993) respectively. The core of the trade-off procedure lies in the search of the weight space generated by preference relations assessed by the DM on key alternatives, which shows the best outcome on one criterion and the worst on the others. The aim of the more recent FITradeoff is to reduce the cognitive effort required from the DM as much as possible by making the procedure work with less information than is required by the standard trade-off procedure.

The importance of reducing the cognitive efforts of the DM is almost unanimously recognized, but all the aforementioned indirect elicitation procedures require more preference information than our proposal. In our case the effort required by the DM is either to select the most preferred alternative among a subset of alternatives or to rank them, and the amount affects the effort level. To the best of our knowledge, no current PROMETHEE-based eliciting approaches fit this requirement.

3. An overview of the PROMETHEE method

Let $\vec{a}_i \in R^n$ be the vector of the values $(a_i)_j$ that the alternative i shows on the criterion j, with $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, n\}$. The outranking relations on each criterion between couples of alternatives are functions of the differences that the alternatives exhibit on criterion j. Each criterion $j \in \{1, \dots, n\}$ is thus associated with a matrix $D_j \in R^{m \times m}$ whose elements $\left(D_j\right)_{ik}$ are given by the difference between the alternatives i and k, that is:

$$\left(D_j\right)_{ik} = (a_i)_j - (a_k)_j \tag{1}$$

Each criterion is also associated with a preference function $f_j: R \to R$, which translates the differences into a 0-1 range according to one of the six standard preference functions reported by Brans and Vincke (1985) and Brans, Marescha and Vincke (1984); a preference function maps $D_j \in R^{m \times m}$ to a preference matrix $P_j \in R^{m \times m}$, $\forall\, j \in \{1, \dots, n\}$, whose elements for $i, k \in \{1, \dots, m\}$ are given by:

$$\left(P_j\right)_{ik} = f_j\left(D_j\right)_{ik} \tag{2}$$

The criteria are then weighted by $\vec{w} \in R^n$, whose elements $(w)_j$, with $j \in \{1, \dots, n\}$, represent the weight given to criterion j. This step enables the aggregation of all the preferences in a single matrix $S \in R^{m \times m}$ given by:

$$S = \sum_{j=1}^{n} (w)_j \cdot P_j \tag{3}$$

For each alternative i, with $i \in \{1, \dots, m\}$, the outgoing $\phi_i^+ \in R$ and incoming $\phi_i^- \in R$ flows are calculated as:

$$\phi_i^+ = \sum_{k=1}^{m} (S)_{ik} \tag{4}$$

$$\phi_i^- = \sum_{k=1}^{m} (S)_{ki} \tag{5}$$

The outgoing flow indicates how much the alternative a_i is preferred over the other alternatives. Conversely the incoming flow indicates how much the other alternatives are preferred over a_i. According to Eqs (4) and (5), PROMETHEE I builds preference, indifference and incomparability relations between alternatives, while PROMETHEE II allows a complete ranking to be attained by calculating the net flow $\phi_i \in R$, which summarizes its preference for each alternative over the others. The net flow is calculated as:

$$\phi_i = \phi_i^+ - \phi_i^- \tag{6}$$

Hence, the alternatives are ranked by decreasing net flows.

In order to help the reader to understand the rationale behind PROMETHEE II, in particular the use of the preference functions, a very simple example is reported below. This is the core of all PROMETHEE-based methods to solve both ranking and sorting problems (see, for instance, Lolli, Ishizaka, Gamberini, Rimini, & Messori, 2015 or Ishizaka & Nemery, 2013).

Suppose we have three alternatives ($m = 3$) evaluated on three criteria ($n = 3$) with the values reported in Table 1. The matrixes D_1, D_2, D_3 contain the differences between the values shown by the alternatives on each criterion (see D_1 in Figure 1). Each criterion is associated with a preference function to translate the said differences into a value in the 0-1 range. For instance, the preference function f_1 (see Figure 2) is such that:

- $f_1(D_1)_{ik} = 0$ if $(D_1)_{ik} < 0$;
- $f_1(D_1)_{ik} = (D_1)_{ik} / \max\limits_{i,k}\{(D_1)_{ik}\}$ if $(D_1)_{ik} \geq 0$, with $\max\limits_{i,k}\{(D_1)_{ik}\} = 17$.

It follows that $(P_1)_{11} = (P_1)_{22} = (P_1)_{33} = (P_1)_{12} = (P_1)_{13} = (P_1)_{23} = 0$, $(P_1)_{21} = 0.65$, $(P_1)_{31} = 1$, and $(P_1)_{32} = 0.35$.

	Criterion 1	Criterion 2	Criterion 3
a_1	24	0.7	100

a_2	35	4	80
a_3	41	1.4	50

Table 1. An example of data for the PROMETHEE application.

$$
\begin{array}{c c}
 & \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\
\begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} & \begin{pmatrix} 0 & -11 & -17 \\ 11 & 0 & -6 \\ 17 & 6 & 0 \end{pmatrix}
\end{array}
$$

Figure 1. The matrix D_1 containing the differences on criterion 1.

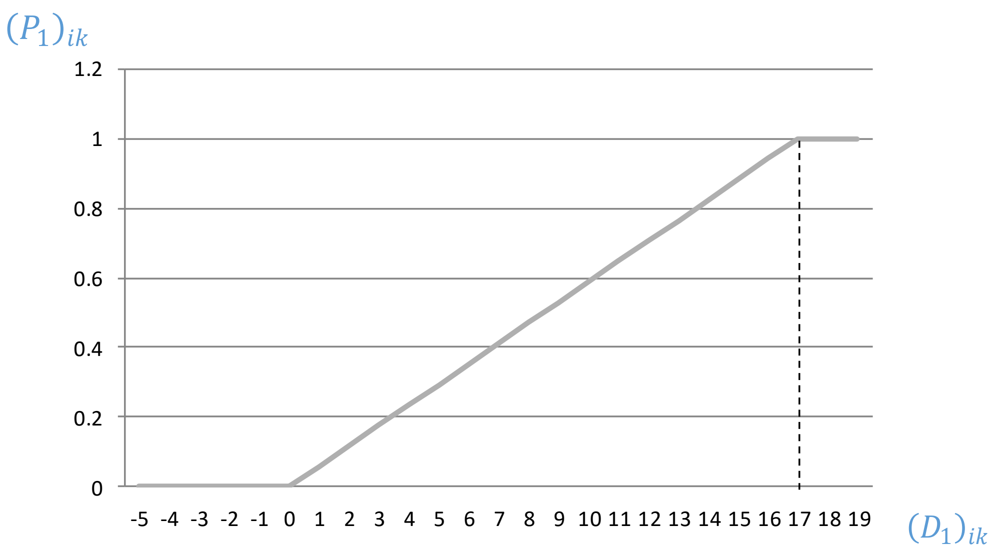

Figure 2. The preference function f_1.

After assigning the weights to criteria, the matrix S is calculated (Eq. (3)), as well as the outgoing, incoming, and net flows for ranking the alternatives.

4. Elicitation approaches

By ‘cognitive efforts’, we mean the information that the DM requires in order to make our model feasible. The goal of the subsequent models is simply to elicit approaches, that is to find the preference structure of DM (i.e. the unknown vector of criteria weights $\vec{w}$), but only by using his/her past choices iteration-by-iteration. At each iteration, preference relations are known only for a subset of alternatives $A \subseteq \{\vec{a}_1, \dots, \vec{a}_m\}$, which are named reference alternatives, and the notation $\vec{a}_i P \vec{a}_k$ means that the alternative i is preferred over the alternative k. It is supposed that the user indicates the ranking of the reference alternatives according to its vector of weights $\vec{w}$, but all the following approaches may also be applied to a case in which the user indicates only the most preferred alternative into A. In the latter case, the cognitive effort required by the DM is likely to be minimal since he/she is only asked for the preferred alternative, which just corresponds to the normal use of the mobile application.

In the following subsections, four optimisation approaches are proposed incrementally with the aim of estimating the weight $\vec{w}$. Starting from a linear model, LM, two quadratic models QM1 and QM2 are proposed to overcome the limitations of LM and QM1, respectively. LM, QM1 and QM2 work in a single-iteration scheme. Finally, the fourth model DDQM2 is proposed to enable QM2 to work iteratively under the scheme reported in Section 1.

4.1 Linear model - LM

The idea of LM consists in searching for the most discriminating weight $\vec{w}$, which therefore maximises the sum of the separation measures e_{ik} between the alternatives a_i and a_k, such that a_i is preferred to a_k. The weight $\vec{w}$ can be estimated by solving the linear optimization problem:

$$max \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k} e_{ik} \tag{7}$$

$s.t.$

$$\sum_{j=1}^{n} (w)_j \cdot \left(\left(P_j\right)_{ik} - \left(P_j\right)_{ki} \right) \geq e_{ik} \quad \forall \{i,k\} : \vec{a}_i P \vec{a}_k \tag{8}$$

$$e_{ik} \geq 0 \quad \forall i,k : \vec{a}_i P \vec{a}_k \tag{9}$$

$$\sum_{j=1}^{n} (w)_j = 1 \tag{10}$$

$$0 \leq (w)_j \leq 1 \quad \forall j \in \{1, \dots, n\} \tag{11}$$

The resulting weight $\vec{w}$ maximizes the distance between pairs of alternatives with known preference relations. Thus, the most discriminating $\vec{w}$ is searched according to the partial information provided by the DM. However, Eq. (7) pushes the least discriminating criteria to obtain null weights.

4.2 Quadratic model – QM1

The second elicitation approach proposed (QM1) operates from the same premises as LM, but tackles practical issues of LM and aims at a higher level of efficiency.

In Eq. (7) each gain e_{ik} is equally weighted, and the optimization algorithm can decrease the value of a single gain if that action increases the others. Empirically some of the gains can decrease drastically and some of the constraints (8) may no longer hold without the optimizer noticing it. In some settings these numerical errors are acceptable, but in our case the violation of a constraint (8) often leads to incorrect ranking, reversing the preference between two alternatives. In the following model, a quadratic objective function with negative Hessian is implemented. The larger the gain, the smaller the advantage obtainable by further increasing it. The resulting gains are more homogenous and this in turn decreases the occurrence of the constraints violation. The objective function can be written as follows:

$$max \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k} e_{ik} - \frac{h}{2} \cdot \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k} e_{ik}^2 \qquad (12)$$

The parameter h can be initially set to 1 and increased by multiplicative steps of 10 after each failed constraint compliance check. Figure 3 illustrates how different values of $h \in \{0, \dots, 5\}$ change the one-dimensional objective function $max\ e - \frac{h}{2} \cdot e^2$, where the crosses mark the optimal values.

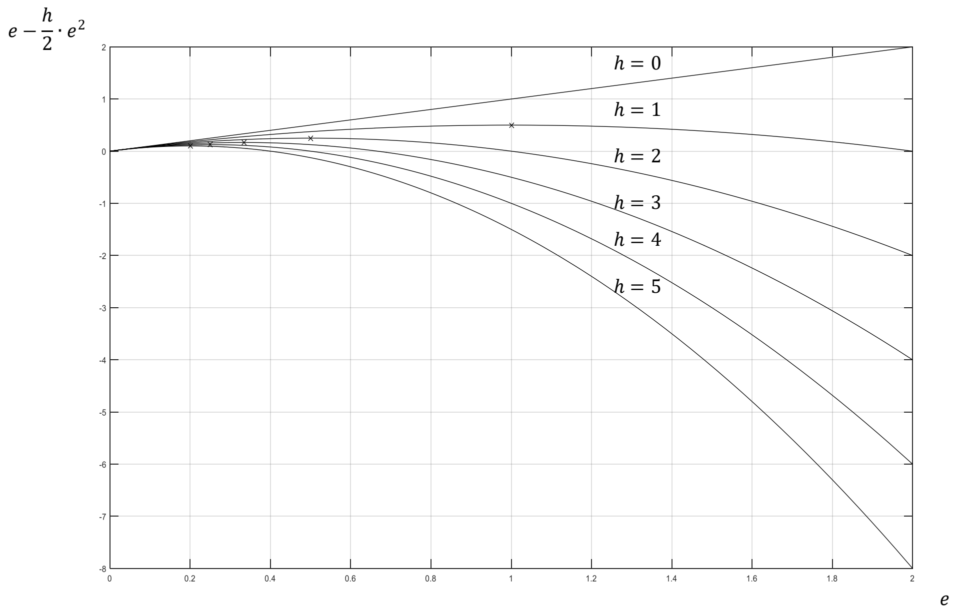

Figure 3. One-dimensional objective functions comparison.

However, both Eq. (7) and Eq. (12) can lead to a weight vector $\vec{w}$ with null components. The optimizer increases the most efficient dimensions of $\vec{w}$ and as the norm 1 of $\vec{w}$ is constrained the less efficient dimensions are disregarded. To avoid this, the objective function is enriched by a second weighted component, i.e., minus half the squared norm of $\vec{w}$, to maximize the weight homogeneity:

$$max \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k} e_{ik} - \frac{h}{2} \cdot \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k} e_{ik}^2 - \frac{g}{2} \sum_{j=1}^{n} (w)_j^2 \quad (13)$$

Where g is a trade-off parameter between objective function minimization and weight homogeneity, as g increases the components of $\vec{w}$ become more and more similar up to an extreme case where they become the same irrespectively of any changes over the gains.

Figure 4 illustrates the maximization of the objective function $-\frac{1}{2}\sum_{j=1}^{2}(w)_j^2$ given a bi-dimensional weight $\vec{w}$ subject to the constraints $\sum_{j=1}^{2}(w)_j = 1$ and $0 \leq (w)_j \leq 1 \; \forall j \in \{1,2\}$. The dot is the optimum, the dotted lines are objective functions contours and the solid line is the constraint.

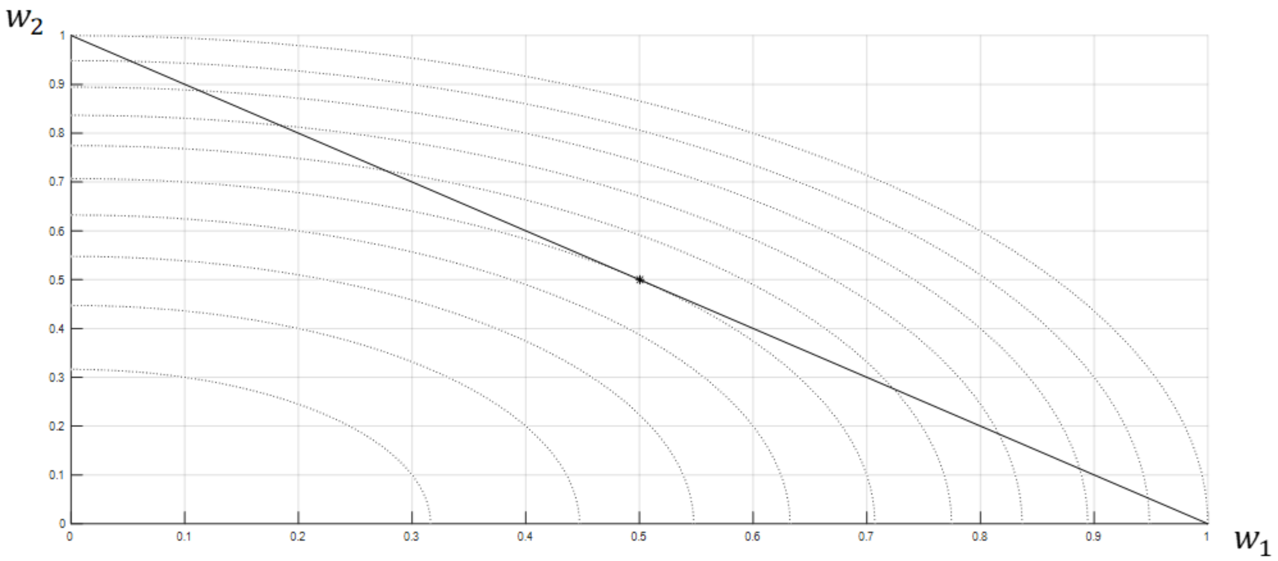

Figure 4. Squared norm maximization with constrained positive weights.

From Figure 4 it is evident that Eq. (13) favours balanced over unbalanced solutions.

If A is strictly ordered, i.e. $\forall \vec{a}_i, \vec{a}_k \in A\ \exists \vec{a}_i P \vec{a}_k | \vec{a}_k P \vec{a}_i$, the gain e_{ik} can be re-defined.

Let $A_p = \{\vec{a}_i P \vec{a}_k : \vec{a}_i \in \{\vec{a}_1, \dots, \vec{a}_m\}, \vec{a}_k \in \{\vec{a}_1, \dots, \vec{a}_m\}\}$ be the set of preferences obtained from the subset A and let $A_{pc} = \{\vec{a}_i P \vec{a}_k : \nexists \vec{a}_l \in A, \vec{a}_i P \vec{a}_l \in A_p, \vec{a}_l P \vec{a}_k \in A_p, \vec{a}_i P \vec{a}_k \in A_p\}$ be the set of preferences between close alternatives. It follows that $A_{pc} \subseteq A_p$, and using A_{pc} instead of A_p to generate the gains produces a smaller number of constraints without violating the preference order. The QM1 model can be formulated as follows:

$$max \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k \in A_{pc}} \left(e_{ik} - \frac{h}{2} e_{ik}^2\right) - \frac{g}{2} \sum_{j=1}^{n} (w)_j^2 \tag{14}$$

$s.t.$

$$\sum_{j=1}^{n} (w)_j \cdot \left(\left(P_j\right)_{ik} - \left(P_j\right)_{ki}\right) \geq e_{ik} \quad \forall \{i,k\} :\ \vec{a}_i P \vec{a}_k \in A_{pc} \tag{15}$$

$$e_{ik} \geq 0 \quad \forall i,k :\ \vec{a}_i P \vec{a}_k \in A_{pc} \tag{16}$$

$$\sum_{j=1}^{n} (w)_j = 1 \tag{17}$$

$$0 \leq (w)_j \leq 1 \quad \forall j \in \{1, \dots, n\} \tag{18}$$

4.3 Quadratic model – QM2

As outlined by Mareschal, De Smet and Nemery (2008), the PROMETHEE method is subject to rank reversal, so that adding/removing some alternatives can result in changes in the relative ranking of those already in place. Both LM and QM1 suffer for this property as they use only the pair-wise comparison between the alternatives a_i and a_k, i.e., a fraction of the net flow, to generate respectively the constraints (8) and (15). Conversely, the difference between the net flows of a_i and a_k in the PROMETHEE II method is also affected by the other alternatives, but this effect is not captured by the constraints. It follows that rank reversal may arise when the net flows of all the reference alternatives are taken into account. To overcome the risk of rank reversal into $\{\vec{a}_1, \dots, \vec{a}_m\}$, QM2 can be articulated as follows:

$$max \sum_{\{i,k\}\,:\,\vec{a}_i P \vec{a}_k \in A_{pc}} \left(e_{ik} - \frac{h}{2} e_{ik}^2\right) - \frac{g}{2} \sum_{j=1}^{n} (w)_j^2 \tag{19}$$

$s.t.$

$$\sum_{j=1}^{n} \sum_{l=1}^{m} (w)_j \cdot \left(\left(P_j\right)_{il} - \left(P_j\right)_{kl}\right) \geq e_{ik} \quad \forall \{i,k\}:\ \vec{a}_i P \vec{a}_k \in A_{pc} \tag{20}$$

$$e_{ik} \geq 0 \quad \forall i,k:\ \vec{a}_i P \vec{a}_k \in A_{pc} \tag{21}$$

$$\sum_{j=1}^{n} (w)_j = 1 \tag{22}$$

$$0 \leq (w)_j \leq 1 \quad \forall j \in \{1, \dots, n\} \tag{23}$$

The use of net flows in Eq. (20) eliminates the risk of rank reversal into $\{\vec{a}_1, \dots, \vec{a}_m\}$ by accounting for all the interactions between alternatives. As this model is more computationally expensive, it is advisable to use QM1 in all contexts where the rank reversal is not an issue, and the most common case is when the preference function is linear.*Data-driven QM2 – DDQM2*

QM2 elicits the weight $\vec{w}$ for a predefined set of alternatives $\vec{a}_1, \dots, \vec{a}_m$ given a subset of alternatives A with known preference relations. In such a single-iteration scheme, the obtained $\vec{w}$ is then used in the PROMETHEE method to rank $\vec{a}_1, \dots, \vec{a}_m$. While the aforementioned eliciting models are suitable for this purpose, they can also be effectively used to elicit $\vec{w}$ given past choices, i.e., iteration by iteration. Hence, starting from QM2, the iterative elicitation procedure DDQM2 can be formulated. The underlying assumption is that the user's choices are coherent with $\vec{w}$ and that $\vec{w}$ does not change over iterations. The issue of inconsistency can be addressed in future research.

To extend QM2 to an iterative procedure, the previously mentioned notation must be expanded. Given a single iteration $(s+1)$, let $A_1, \dots, A_s$ be the s strictly ordered sets of alternatives coming from the past s iterations, not necessarily with the same cardinalities, and referring to the past decisions of the DM; for each $A_q \epsilon \{A_1, \dots, A_s\}$ a corresponding set of preferences $A_{p\_q} = \{\vec{a}_{i\_q} P \vec{a}_{k\_q} \colon \vec{a}_{i\_q} \in A_q, \vec{a}_{k\_q} \in A_q\}$ can be obtained with $A_{pc\_q} = \{\vec{a}_{i\_q} P \vec{a}_{k\_q} \colon \nexists \vec{a}_{l\_q} \in A_q, \vec{a}_{i\_q} P \vec{a}_{l\_q} \in A_{p\_q}, \vec{a}_{l\_q} P \vec{a}_{k\_q} \in A_{p\_q}, \vec{a}_{i\_q} P \vec{a}_{k\_q} \in A_{p\_q}\}$ being the set of preferences between close alternatives. The remaining notation can also be expanded defining $\vec{a}_{i\_q} \in A_q$ as a member of the set A_q and $P_{1\_q}, \dots, P_{n\_q}$ as the set preference matrices. Using the new notation, the iterative version of QM2 can be written as:

$$max \sum_{q=1}^{s} \sum_{\{i\_q, k\_q\} : \vec{a}_{i\_q} P \vec{a}_{k\_q} \in A_{pc\_q}} \left(e_{ik\_q} - \frac{h}{2} e_{ik\_q}^2 \right) - \frac{g}{2} \sum_{j=1}^{n} (w)_j^2 \tag{24}$$

$s.t.$

$$\sum_{j=1}^{n} \sum_{l=1}^{m\_q} (w)_j \cdot \left(\left(P_{j\_q}\right)_{il} - \left(P_{j\_q}\right)_{kl} \right) \geq e_{ik\_q} \quad \forall \{i\_q, k\_q\} : \vec{a}_{i\_q} P \vec{a}_{k\_q} \in A_{pc\_q} \quad \forall q = 1, \dots, s \tag{25}$$

$$e_{ik\_q} \geq 0 \quad \forall i, k : \vec{a}_{i\_q} P \vec{a}_{k\_q} \in A_{pc\_q} \quad \forall q = 1, \dots, s \tag{26}$$

$$\sum_{j=1}^{n} (w)_j = 1 \tag{27}$$

$$0 \leq (w)_j \leq 1 \quad \forall j \in \{1, \dots, n\} \tag{28}$$

The vector $\vec{w}$ achieved allows to rank the alternatives available at the current iteration $(s+1)$, among which a sample is extracted (randomly or according to their ranking) and ordered by the

DM, thereby generating the new ordered set A_{s+1} to be used in the subsequent iteration $(s+2)$, and so on. Hence, all the past sets of alternatives are accounted for to elicit $\vec{w}$ iteratively. If the preferences are coherent both intra and inter iterations, $\vec{w}$ is expected to converge by increasing the number of past ordered sets iteration by iteration.

5. Experimental setting

LUME PlannER® is an application designed to support tourists in planning their own sustainable routes in the Emilia Romagna region. Its development was co-funded by the European Regional Development Fund 2014-2020, in cooperation with the Regional Secretariat for Emilia Romagna of the Ministry of Cultural Heritage and Activities and Tourism, and the Institute for the Cultural and Natural Artistic Heritage of the Emilia-Romagna Region (IBC-Emilia Romagna). This strategic industrial research project is aimed at smart specialization strategy priority areas, to support both the regional and the local tourism industries. The project acts as an aggregator tool, and gathers and processes tourism-related data, then matches them with the tourists' interests, needs and constraints.

The interface of LUME PlannER® enables the users to select cultural attractions (e.g., historic sites, museums and art and cultural events) and then provides a plan for visiting the selected attractions. If only the total distance covered is taken into account, the problem may be viewed as a multimodal shortest path problem. The application can also generate more feasible solutions (different routes and modes of transport) that can be evaluated through a multicriteria perspective. Without loss of generality, five evaluation criteria are considered: distance, cost, waiting time, environmental sustainability (e.g., carbon dioxide emissions reduction) and expected traffic congestion in route and at the attractions. All the values related to the generated alternatives from the criteria are objective and exactly known. As predicted, each generated alternative may significantly differ from the others in terms of criteria, but the user will select only one of them. Thus a multicriteria selection problem arises, in which the selected alternative implicitly reflects the priorities that the user gives to the evaluation criteria. Supposing that no information on the user's needs is available at the first iteration, the weights for assigning the evaluation criteria are completely unknown. Once the first selection is made, which is translated into the pairwise comparisons between the selected alternative and the remaining ones, information on the user's needs is then available for eliciting the criteria weights; these weights will be used for ranking the alternatives generated at the next iteration and so on. To increase the usability of the application, only the subsets of the three top-ranked alternatives are shown to the user at each iteration, from which he/she either selects the most preferred one or provides their ranking.

The proposed models are validated using three experiments:

1. LM is applied to prove the issues of constraint violations and criteria with null weights. The aim is therefore to justify the introduction of the quadratic models (i.e., QM1, QM2 and DDQM2).
2. QM1 is applied to the same scenarios to confirm it can overcome the weaknesses of LM.
3. DDQM2 is applied to generated data to analyse the convergence properties.

Experiments 1 and 2 refer to non-iterative scenarios, so the elicitation is performed in only one step, while experiment 3 is to validate the iterative DDQM2 in the real setting in which the application should operate.

5.1 Data collection

A questionnaire was completed by 1000 users, to obtain as many 5-dimensional vectors of weights $\vec{w}$ as possible to assign to the evaluation criteria. We used the AHP-based pairwise comparison among criteria on a scale of 1-9 and then used the eigenvector method in the distributive mode to calculate weights that each user assigns to the five criteria (Saaty, 1980). As predicted, some users provided inconsistent judgments (i.e., with consistency ratios higher than the standard threshold value of 0.1). This phenomenon must be checked carefully in AHP-based approaches because it can invalidate the achieved results, but not in our case. In fact, the weight vectors from inconsistent users do not impair the experimental results as our goal is to merely validate the eliciting models.

5.2 Experiments 1 and 2

Experiments 1 and 2 use the aforementioned 1000 vectors of weights adding up to 1 (5-dimensional vectors named $\vec{w}$) and, per each vector of weights, 100 alternatives provide randomly generated values on the 5 criteria. Then 1000 PROMETHEE-based rankings of alternatives are achieved by using the respective $\vec{w}$ and considering linear preference functions for all the criteria with indifference and preference thresholds equal to zero and the maximum difference between alternatives, respectively.

In the first experiment, the pairwise holistic preferences are separately fed to LM, as outlined in Eqs. (7), (8), (9) and (11), and a complete ranking of the alternatives is also provided.

The goodness of the 1000 vectors of weights $\vec{w}_{result}$ that result from launching LM is evaluated by introducing the measure *performance*, defined into the 0-1 range, that computes the cosine of the angle between $\vec{w}$ and $\vec{w}_{result}$ as follows:

$$performance = \frac{\vec{w}' \cdot \vec{w}_{result}}{\|\vec{w}\| \cdot \|\vec{w}_{result}\|} \tag{29}$$

where $\vec{w}'$is the transposed vector of $\vec{w}$. The smaller the angle between $\vec{w}$ and $\vec{w}_{result}$, the bigger the cosine and the smaller the gap between $\vec{w}$ and $\vec{w}_{result}$. If $\vec{w}_{result}$ provides the same ranking obtained by using $\vec{w}$, then this is memorized.

In the second experiment, a similar procedure is applied, using QM1 as outlined in (14), (15), (16), (17) and (18). The parameter $\frac{g}{2}$ is set to $\frac{999}{5}$, i.e., the ratio between the preference constraint and weight constraints in the objective function, to balance the two concurring objectives. The parameter h is updated, as specified in Section 4.2, after each optimization if the obtained ranking diverges from that obtained by the PROMETHEE method previously computed by using $\vec{w}$.

5.3 Experiment 3

The third experiment is designed to reproduce an iterative setting, similar to the real-world context in which the application operates. For the whole experiment a single vector $\vec{w}$ (i.e., 1 user) and 200 groups, each one made up of three reference alternatives, are used. Each alternative is defined by five values, one for each criterion, adding up to 1, and each group is ranked using the standard PROMETHEE II with $\vec{w}$ as the vector of weights, indifference thresholds equal to zero, and preference thresholds equal to the maximum difference between alternatives.

The ranked groups are fed one by one to DDQM2, as outlined in Eqs. (24), (25), (26), (27) and (28), maintaining in each iteration the past constraints and objective function elements. The parameter $\frac{g}{2}$ is set to $\frac{2}{5}$, the ratio between preference constraint and weight constraints in the objective function, to balance the two concurring objectives. Once a group has been fed to DDQM2 it will be available to the model in the subsequent iterations as well.

The performance measure (Eq. (29)) is computed for each iteration with the aim of evaluating the convergence ability of DDQM2 to $\vec{w}$ over the iterations.

6. Experimental results

The result of experiments 1 and 2 are outlined in Table 2, where the mean, the median and the standard deviation of *performance* (Eq. (29)) on the 1000 vectors of weights are reported along with the PROMETHEE error ratio and the null weight ratio. The former is the ratio between wrong rankings and 1000 and the latter is the ratio between $\vec{w}_{result}$ with at least one null weight and 1000.

	experiment 1	experiment 2
performance mean	0.972455678	0.935910074
performance median	0.988560962	0.946292819
performance standard deviation	0.039983414	0.052105099

PROMETHEE error ratio	0.879	0
null weight ratio	1	0

Table 2. Result of experiments 1 and 2.

Figure 5 plots the frequency histogram of the performance obtained in the first scenario and Figure 6 the frequency histogram of the precision obtained in the second scenario.

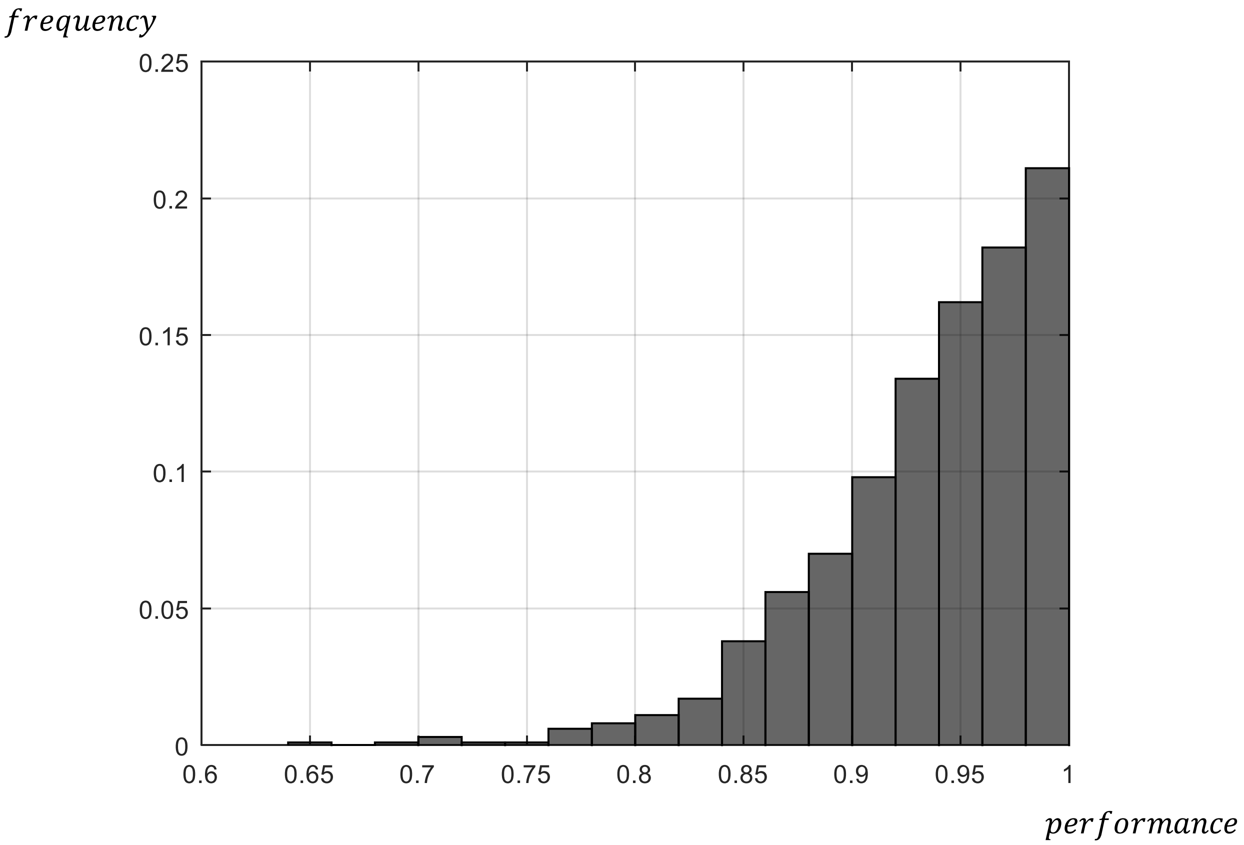

Figure 5. Performance distribution of experiment 1.

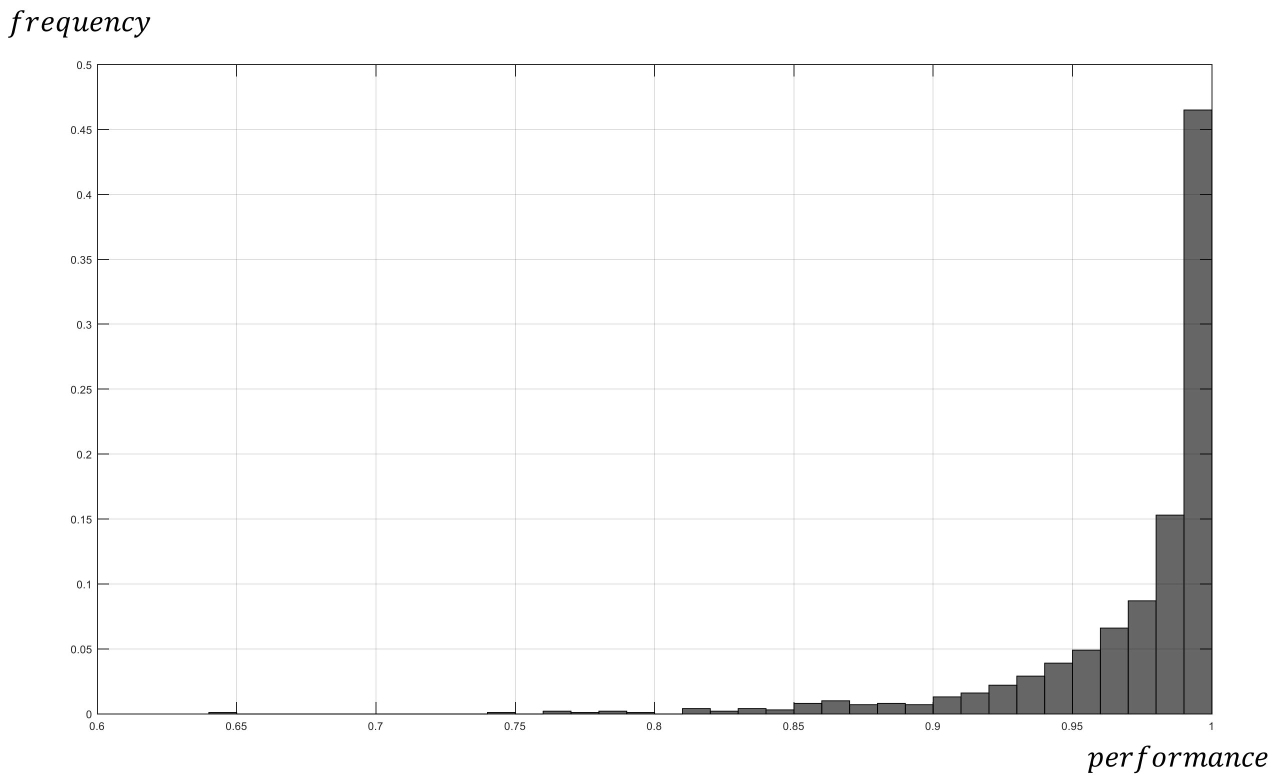

Figure 6. Performance distribution of experiment 2.

From Table 2 it follows that the performance mean and the performance median of experiment 1 are higher than the ones of experiment 2, i.e. QM1 is slightly less precise than LM with a performance slightly more spread than that of LM, probably due to the regularization favouring a balanced weight. However, the ranking obtained using the vectors of weights $\vec{w}_{result}$ elicited by QM1 is equivalent to the one obtained using the vectors $\vec{w}$, but this is not true for LM. For a problem of this scale LM produces at least one inversion 88% of the times. In addition, LM always reaches $\vec{w}_{result}$ with at least one null weight, while QM1 never presents near null (an absolute value smaller than 10^{-10}) weights. In other words, LM pays its higher precision with a higher PROMETHEE error ratio as well as a higher null weight ratio.

The results of experiment 3 are outlined in Figures 7 and 8.

Figure 7 tracks the increase in performance generated over the iterations by DDQM2. The performance can be observed to sharply increase from the initial 0.8391 up to 0.9998 after the first 4 iterations, and then decreases slightly converging to 0.9998 in the long run. The slight decrease after the first increment is probably a result of the objective function prioritizing a balanced solution over optimal ones, as described in Section 3.

Figure 8 measures the smallest weight into $\vec{w}_{result}$, given $2.877 \cdot 10^{-2}$ as the smallest one into $\vec{w}$; it starts from $2.116 \cdot 10^{-12}$ for a single iteration and is in the range of $3 \cdot 10^{-2} \div 4 \cdot 10^{-2}$ after about 80 iterations and up to 200 iterations. This is clearly the effect of the balancing factor in the objective function of DDQM2, which favours balanced solutions that penalise the performance measure.

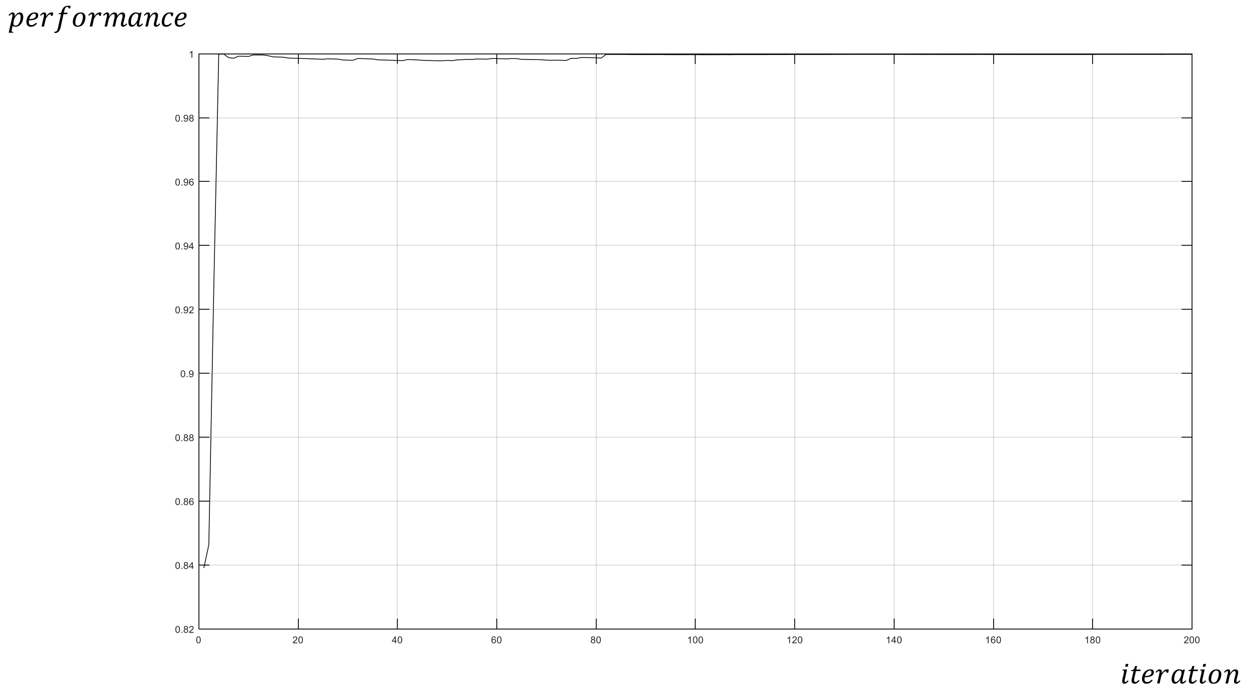

Figure 7. The behaviour of the performance achieved by DDQM2 in experiment 3.

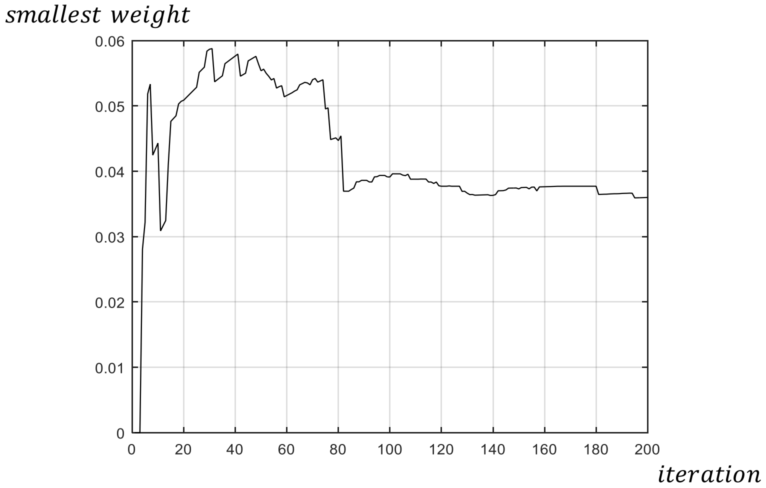

Figure 8. The smallest weight obtained over the iterations by DDQM2 in experiment 3.

7. Conclusions

Indirect procedures aimed at eliciting the criteria weights in multicriteria problems appear to be the most suitable strategies for inferring the DM’s preference structure, while reducing his/her cognitive efforts as much as possible. This is a rational goal when designing mobile applications that work on an underlying multicriteria problem, when usability is strictly related to the efforts required from the users.

Different indirect eliciting models are proposed in this study, which can then be applied in iterative form to a mobile application working on an underlying multicriteria selection problem. The models were progressively introduced, beginning with a linear model aimed at searching for the most discriminating vector of weights, if the information provided by the user is the ranking of a subset of reference items. However, the linear model has issues that the subsequent models are able to address. An iterative quadratic model is finally proposed to solve data-driven weight eliciting problems, where the objective is not to rank known alternatives with unknown criteria weights but to elicit unknown criteria weights only after analysing the past decisions provided by the user.

An experimental validation is used to confirm the robustness of the proposed models and to map their performance both in a one-step version and in a real-world iterative setting for the mobile application. Overall, the quadratic eliciting methods were found to be slightly less precise but generally more robust than the linear model, trading closeness to the correct weight for avoidance of constraints violations.

Further research will be aimed at analysing both the linear and the quadratic models in a context of inconsistently ordered alternatives. The DDQM2 is believed to be robust against these problems if its constraints are correctly managed. Alternatively, interactive or automatic procedures for managing inconsistencies could be introduced. Finally, specialized machine-learning approaches will be developed and compared with the DDQM2. Machine-learning procedures could trade off precision over the past decisions' outcomes for precision over future decisions without overly distorting the results as in the linear model. However, such methodologies would be harder to implement and potentially more reliant on meta-parameters.

Acknowledgements

The research has been co-funded by the European Regional Development Fund (ERDF) - ROP of the Emilia-Romagna Region (IT), within the framework of the LUME-PlannER project (POR-FESR 2014-2020, DGR 774/2015 - CUP E82I15002660002).

References

Alemi-Ardakani, M., Milani, A. S., Yannacopoulos, S., & Shokouhi, G. (2016). On the effect of subjective, objective and combinative weighting in multiple criteria decision making: A case study on impact optimization of composites. *Expert Systems with Applications*, *46*, 426–438. https://doi.org/10.1016/j.eswa.2015.11.003

Angilella, S., Corrente, S., Greco, S., & Słowiński, R. (2016). Robust Ordinal Regression and Stochastic Multiobjective Acceptability Analysis in multiple criteria hierarchy process for the Choquet integral preference model. *Omega (United Kingdom)*, *63*, 154–169. https://doi.org/10.1016/j.omega.2015.10.010

Angilella, S., Greco, S., & Matarazzo, B. (2010). Non-additive robust ordinal regression: A multiple criteria decision model based on the Choquet integral. *European Journal of Operational Research*, *201*(1), 277–288. https://doi.org/10.1016/j.ejor.2009.02.023

Barton, K. C. (2015). Elicitation techniques: Getting people to talk about ideas they dont usually talk about. *Theory and Research in Social Education*, *43*(2), 179–205. https://doi.org/10.1080/00933104.2015.1034392

Brans, J. P., Mareschal, B., & Vincke, P. (1984). PROMETHEE: A New Family of Outranking Methods in Multicriteria Analysis. In *Operational Research* (pp. 477–490).

Brans, J. P., & Vincke, P. (1985). A Preference Ranking Organisation Method: (The PROMETHEE Method for Multiple Criteria Decision-Making). *Management Science*, *31*(6), 647–656. https://doi.org/10.1287/mnsc.31.6.647

Corrente, S., Figueira, J. R., & Greco, S. (2014). The SMAA-PROMETHEE method. *European Journal of Operational Research*, *239*(2), 514–522. https://doi.org/10.1016/j.ejor.2014.05.026

Corrente, S., Greco, S., & SŁowiński, R. (2013). Multiple Criteria Hierarchy Process with ELECTRE and PROMETHEE. *Omega (United Kingdom)*, *41*(5), 820–846. https://doi.org/10.1016/j.omega.2012.10.009

De Almeida, A. T., De Almeida, J. A., Costa, A. P. C. S., & De Almeida-Filho, A. T. (2016). A new method for elicitation of criteria weights in additive models: Flexible and interactive tradeoff. *European Journal of Operational Research*, *250*(1), 179–191. https://doi.org/10.1016/j.ejor.2015.08.058

de Almeida Filho, A. T., Clemente, T. R. N., Morais, D. C., & de Almeida, A. T. (2018). Preference modeling experiments with surrogate weighting procedures for the PROMETHEE method. *European Journal of Operational Research*, *264*(2), 453–461. https://doi.org/10.1016/j.ejor.2017.08.006

De Matteis, D., Ishizaka, A., & Resce, G. (2018). The ‘postcode lottery’ of the Italian public health bill analysed with the hierarchy Stochastic Multiobjective Acceptability Analysis. *Socio-Economic Planning Sciences*. https://doi.org/10.1016/j.seps.2017.12.001

Dias, L. C., & Mousseau, V. (2005). Inferring Electre’s veto-related parameters from outranking examples. *European Journal of Operational Research*, *170*(1), 172–191. https://doi.org/10.1016/j.ejor.2004.07.044

Dickinson, J. E., Ghali, K., Cherrett, T., Speed, C., Davies, N., & Norgate, S. (2014). Tourism and the smartphone app: capabilities, emerging practice and scope in the travel domain. *Current Issues in Tourism*, *17*(1), 84–101. https://doi.org/10.1080/13683500.2012.718323

Figueira, J., & Roy, B. (2002). Determining the weights of criteria in the ELECTRE type methods with a revised Simos’ procedure. *European Journal of Operational Research*, *139*(2), 317–326. https://doi.org/10.1016/S0377-2217(01)00370-8

Greco, S., Ishizaka, A., Matarazzo, B., & Torrisi, G. (2018). Stochastic multi-attribute acceptability analysis (SMAA): an application to the ranking of Italian regions. *Regional Studies*, *52*(4), 585–600. https://doi.org/10.1080/00343404.2017.1347612

Greco, S., Kadziński, M., Mousseau, V., & Słowiński, R. (2011). ELECTREGKMS: Robust ordinal regression for outranking methods. *European Journal of Operational Research*, *214*(1), 118–135. https://doi.org/10.1016/j.ejor.2011.03.045

Greco, S., Kadziński, M., & Słowiński, R. (2011). Selection of a representative value function in robust multiple criteria sorting. *Computers & Operations Research*, *38*(11), 1620–1637. https://doi.org/10.1016/j.ejor.2011.09.032

Greco, S., Mousseau, V., & Słowiński, R. (2008). Ordinal regression revisited: Multiple criteria ranking using a set of additive value functions. *European Journal of Operational Research*, *191*(2), 416–436. https://doi.org/10.1016/j.ejor.2007.08.013

Greco, S., Mousseau, V., & Słowiński, R. (2010). Multiple criteria sorting with a set of additive value functions. *European Journal of Operational Research*, *207*(3), 1455–1470. https://doi.org/10.1016/j.ejor.2010.05.021

Hickey, A. M., & Davis, A. M. (2003). Elicitation technique selection: How do experts do it? In *Proceedings of the IEEE International Conference on Requirements Engineering* (Vol. 2003–Janua, pp. 169–178). https://doi.org/10.1109/ICRE.2003.1232748

Hokkanen, J., Lahdelma, R., Miettinen, K., & Salminen, P. (1998). Determining the implementation order of a general plan by using a multicriteria method. *Journal of Multi-Criteria Decision Analysis*, *7*(5), 273–284. https://doi.org/10.1002/(SICI)1099-1360(199809)7:5<273::AID-MCDA198>3.0.CO;2-1

Ishizaka, A., & Nemery, P. (2013). A multi-criteria group decision framework for partner grouping when sharing facilities. *Group Decision and Negotiation*, *22*(4), 773–799. https://doi.org/10.1007/s10726-012-9292-8

Jacquet-Lagreze, E., & Siskos, J. (1982). Assessing a set of additive utility functions for multicriteria decision-making, the UTA method. *European Journal of Operational Research*, *10*(2), 151–164. https://doi.org/10.1016/0377-2217(82)90155-2

Kadziński, M., Greco, S., & Słowiński, R. (2012). Extreme ranking analysis in robust ordinal regression. *Omega*, *40*(4), 488–501. https://doi.org/10.1016/j.omega.2011.09.003

Kadziński, M., & Tervonen, T. (2013). Robust multi-criteria ranking with additive value models and holistic pair-wise preference statements. *European Journal of Operational Research*, *228*(1), 169–180. https://doi.org/10.1016/j.ejor.2013.01.022

Kaliszewski, I., & Podkopaev, D. (2016). Simple additive weighting - A metamodel for multiple criteria decision analysis methods. *Expert Systems with Applications*, *54*, 155–161. https://doi.org/10.1016/j.eswa.2016.01.042

Keeney, R. L., & Raiffa, H. (1993). *Decisions with multiple objectives–preferences and value tradeoffs*. *Behavioral Science* (Vol. 39). https://doi.org/10.1002/bs.3830390206

Lahdelma, R., Hokkanen, J., & Salminen, P. (1998). SMAA - Stochastic multiobjective acceptability analysis. *European Journal of Operational Research, 106*(1), 137–143. https://doi.org/10.1016/S0377-2217(97)00163-X

Lolli, F., Ishizaka, A., Gamberini, R., Rimini, B., & Messori, M. (2015). FlowSort-GDSS - A novel group multi-criteria decision support system for sorting problems with application to FMEA. *Expert Systems with Applications*, *42*(17–18), 6342–6349. https://doi.org/10.1016/j.eswa.2015.04.028

Mareschal, B. (1988). Weight stability intervals in multicriteria decision aid. *European Journal of Operational Research*, *33*(1), 54–64. https://doi.org/10.1016/0377-2217(88)90254-8

Mareschal, B., De Smet, Y., & Nemery, P. (2008). Rank reversal in the PROMETHEE II method: Some new results. In *2008 IEEE International Conference on Industrial Engineering and Engineering Management* (pp. 959–963). IEEE. https://doi.org/10.1109/IEEM.2008.4738012

Mousseau, V., & Slowinski, R. (1998). Inferring an ELECTRE TRI Model from Assignment Examples. *Journal of Global Optimization*, *12*(2), 157–174. https://doi.org/10.1023/A:1008210427517

Riabacke, M., Danielson, M., & Ekenberg, L. (2012). State-of-the-art prescriptive criteria weight elicitation. *Advances in Decision Sciences*, *2012*. https://doi.org/10.1155/2012/276584

Saaty, T. L. (1980). *The Analytic Hierarchy Process*. *Decision Analysis*. Retrieved from http://www.expertchoice.com/markets/index.html#AHP

Simos, J. (1990). *Évaluer l'impact sur l'environnement. Une approche originale par l'analyse multicritère et la négotiation [Environmental impact assessment. An original approach for multi-criteria analysis and negociation]*. *Géographie physique et Quaternaire*.

https://doi.org/10.7202/032939ar

Tervonen, T., Figueira, J. R., Lahdelma, R., Dias, J. A., & Salminen, P. (2009). A stochastic method for robustness analysis in sorting problems. *European Journal of Operational Research*, *192*(1), 236–242. https://doi.org/10.1016/j.ejor.2007.09.008

Vetschera, R. (2017). Deriving rankings from incomplete preference information: A comparison of different approaches. *European Journal of Operational Research*, *258*(1), 244–253. https://doi.org/10.1016/j.ejor.2016.08.031

Weber, M., & Borcherding, K. (1993). Behavioral influences on weight judgments in multiattribute decision making. *European Journal of Operational Research*. https://doi.org/10.1016/0377-2217(93)90318-H