

Use of the GAI model in multi-criteria decision making: inconsistency handling, interpretation

Christophe Labreuche¹ Michel Grabisch²

¹Thales Research & Technology

²Paris School of Economics, Université Paris I - Panthéon-Sorbonne

Abstract

This paper is devoted to the use of the GAI (Generalized Additive Independence) model in a Multi-Criteria Decision Making context. We first discuss on some new conditions (concerning the sign and monotonicity) to add on the terms appearing in a GAI model. Secondly, we propose some algorithms to propose the learning examples to change or remove, together with an explanation of this, when there are inconsistencies in the learning data. Finally, we propose some importance and interaction indices to interpret a GAI model.

Keywords: Multi-Criteria Decision Making, GAI model, Choquet integral, inconsistency, interpretation

1. Introduction

Multi-Criteria Decision Making (MCDM) aims at representing the preferences of a Decision Maker (DM) regarding how to compare some options on the basis of their values on several attributes. The preferences of the DM can be projected to each attribute separately. Depending on the type assumptions on these preferences over each attribute, two lines of MCDM model can be defined. In the first one (called *attribute-decomposable*), the overall assessment of the options can be decomposed as an aggregation function applied to partial utility functions on each attribute. This representation implies some commensurability among criteria in the sense that the partial utility functions return an assessment in the common evaluation scale (e.g. $[0, 1]$ representing a satisfaction degree). The simplest model of this form uses the weighted sum model as an aggregation function. In the second approach (called *additive-decomposable*), there are still some utility functions over the criteria, but it is not assumed that they return a commensurate evaluation, and one assumes some additivity in the overall utility. The most well-known model is the additive utility.

The previous examples of models are both linear. It has been acknowledged that such a model cannot represent many real-life decision strategies called *interaction* among criteria. In the line of the attribute-decomposable models, this has led to the use of the Choquet integral [1]. It has the ability to represent various important phenomena such as

veto, favor, complementarity among criteria, among others. In the line of additive-decomposable models, the GAI (Generalized Additive Independence) model has been designed as a generalization of the additive utility model [2, 3]. The GAI model has recently increasing interest in the MCDM community [2, 3, 4, 5]. References [4, 5] are interested in learning a GAI model using linear programming.

We also consider the problem of using the GAI model in a MCDM context. There are two major phases in MCDM: learning the GAI model and then explaining the model so-obtained. Our first contribution is to provide a representation of 2-additive GAI models (GAI models where interaction is limited to pairs of attributes). In particular, we show that it is sufficient to consider only positive terms in the expression of the GAI model. In the learning context, we extend the approaches provided in [4, 5] by providing in particular methods to handle inconsistent preferential information provided by the user and explain these inconsistency. We provide a new approach for explaining an inconsistency, based on the Farkas lemma [6, pages 28-29]. In the context of explaining the GAI model, we propose importance and interaction indices relevant for the GAI model. These indices are borrowed from what is used for the Choquet integral.

2. GAI model

We are given a set of n attributes indexed by $N = \{1, \dots, n\}$. Each attribute $i \in N$ is represented by a set X_i which can be discrete or continuous (an interval). The alternatives are characterized by a value on each attribute, and are thus represented by an element of $X = X_1 \times \dots \times X_n$. We aim to represent the overall assessment of a decision maker

$$U : X \rightarrow \mathbb{R}.$$

Many utility models can take this form. There are basically two classes of utility models. The first ones are called *attribute-decomposable* [7] and take the form $U(x) = F(u_1(x_1), \dots, u_n(x_n))$, where the u_i 's $: X_i \rightarrow \mathbb{R}$ are called the *utility functions* (also called *value functions*) and $F : \mathbb{R}^n \rightarrow \mathbb{R}$ is an *aggregation function*. Examples of aggregation functions are the weighted sum or the Choquet integral [8, 1, 9]. The Choquet integral is a versatile aggregation function able to capture various decision strate-

gies representing interaction among criteria. However, the decomposable form forbid this model from representing any type of preferences (see [10] to have some examples of preferences not representable by a decomposable model based on a Choquet integral). The main assumption for this model is that the partial utility functions u_i return evaluations in the same scale. This means that if $u_i(x_i) = u_j(x_j)$, then value x_i on attribute X_i has the same satisfaction/attractiveness as value x_j on attribute X_j . This strong assumption is called *commensurability*.

An alternative class of utility models does not need the commensurability assumption. The most well-known model of this class is the *additive utility* model [7]

$$U(x) = \sum_{i \in N} u_i(x_i) \quad (1)$$

where $u_i : X_i \rightarrow \mathbb{R}$. This class is called *additive-decomposable*. Unlike the decomposable model where the utility functions need to be commensurate, such an assumption is not required with the additive utility model. It is apparent that the weighted sum in the attribute-decomposable model can be put in the form of (1). This model has been generalized to allow some interaction among criteria – under the name of the GAI (Generalized Additive model) [2, 3]. The GAI model takes the form of the sum of utilities over subsets of attributes:

$$U(x) = \sum_{S \in \mathcal{S}} u_S(x_S) \quad (2)$$

where \mathcal{S} is a set of subsets of N , $X_S = \prod_{i \in S} X_i$, $x_S \in X_S$ is the restriction of x over attributes in coalition S , and

$$u_S : X_S \rightarrow \mathbb{R}.$$

Set \mathcal{S} contains all subsets of attributes that interact one another. Hence the additive model (1) is a particular case of the GAI model where \mathcal{S} is composed of only singletons.

We assume that we are given a preference relation \succsim_i over each attribute i . We denote by \succ_i and \sim_i the asymmetric and symmetric parts of \succsim_i respectively. Utility U is assumed to fulfill the following monotonicity conditions:

$$\forall x, y \in X \text{ with } y_i \succsim_i x_i \ \forall i \in N, \quad U(y) \geq U(x) \quad (3)$$

3. Background on the Choquet integral

3.1. Basic definitions

Definition 1 A fuzzy measure [11] or capacity [8] on N is a set function $\mu : 2^N \rightarrow \mathbb{R}$ satisfying

- $A \subseteq B \Rightarrow \mu(A) \leq \mu(B)$,
- $\mu(\emptyset) = 0$, $\mu(N) = 1$.

Capacity μ is said to be *non-normalized* (it is also called a *game*) if condition $\mu(N) = 1$ is relaxed. The Möbius transform (see e.g. [12]) of μ is defined by

$$m^\mu(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \mu(B). \quad (4)$$

Reciprocally, μ can be recovered from the Möbius transform by

$$\mu(A) = \sum_{B \subseteq A} m^\mu(B). \quad (5)$$

The Choquet integral of $a \in \mathbb{R}^N$ w.r.t. capacity μ (also called the *Lovász extension*) is defined by [8]

$$C_\mu(a) = \sum_{A \subseteq N} m^\mu(A) \cdot \bigwedge_{i \in A} a_i, \quad \forall a \in \mathbb{R}_+^n \quad (6)$$

where m^μ is the Möbius transform of μ , and \wedge is the min operator. An equivalent expression in terms of the capacity μ is

$$C_\mu(a) = \sum_{i=1}^n (a_{\tau(i)} - a_{\tau(i-1)}) \mu(\{\tau(i), \dots, \tau(n)\}), \quad (7)$$

where $a_{\tau(0)} := 0$ and τ is a permutation on N such that $a_{\tau(1)} \leq a_{\tau(2)} \leq \dots \leq a_{\tau(n)}$.

The *discrete derivative* of μ w.r.t. a coalition $P \subseteq N$ at a coalition S , with $S \subseteq N \setminus P$, is recursively defined by $\Delta_P \mu(S) := \Delta_i [\Delta_{P \setminus \{i\}} \mu](S)$ for all $i \in P$ [13]. One has when $S \cap P = \emptyset$ [13]

$$\Delta_P \mu(S) = \sum_{T \subseteq P} (-1)^{|P| - |T|} \mu(S \cup T).$$

The importance index, known as the *Shapley importance index* is defined by [14]

$$\phi^\mu(i) := \sum_{K \subseteq N \setminus i} \frac{(n - |K| - 1)! |K|!}{n!} \Delta_i \mu(K) \quad (8)$$

The interaction index for a coalition $\emptyset \neq A \subseteq N$ of criteria is [15]:

$$I^\mu(A) := \sum_{K \subseteq N \setminus A} \frac{(n - k - |A|)! k!}{(n - |A| + 1)!} \Delta_A \mu(K) \quad (9)$$

This definition is due to Murofushi and Soneda [16] for pairs of criteria. We also note that $I^\mu(\{i\}) = \phi^\mu(i)$. We have [17]

$$I^\mu(A) = \sum_{B \subseteq N \setminus A} \frac{1}{|B| + 1} m^\mu(A \cup B). \quad (10)$$

3.2. Two-additive model

Definition 2 [15] Let $k \in \{1, \dots, n-1\}$. A capacity μ is said to be *k-additive* if $I^\mu(A) = 0$ whenever $|A| > k$, and there exists some $A \subseteq N$ with $|A| = k$ such that $I^\mu(A) \neq 0$.

An equivalent definition is that $m^\mu(A) = 0$ whenever $|A| > k$, and there exists some $A \subseteq N$ with $|A| = k$ such that $m^\mu(A) \neq 0$.

The Choquet integral can be expressed using I instead of μ in a very instructive way when the measure is 2-additive [18]:

$$\begin{aligned} C_\mu(a_1, \dots, a_n) &= \sum_{I^\mu(\{i,j\}) > 0} (a_i \wedge a_j) I^\mu(\{i,j\}) \\ &+ \sum_{I^\mu(\{i,j\}) < 0} (a_i \vee a_j) |I^\mu(\{i,j\})| \\ &+ \sum_{i=1}^n a_i (\phi^\mu(i) - \frac{1}{2} \sum_{j \neq i} |I^\mu(\{i,j\})|) \end{aligned} \quad (11)$$

for all $(a_1, \dots, a_n) \in \mathbb{R}^n$, with the property that $\phi^\mu(i) - \frac{1}{2} \sum_{j \neq i} |I^\mu(\{i,j\})| \geq 0$ for all i .

Consider a 2-additive capacity μ . From (4), we note that

$$\begin{aligned} m^\mu(\{i,j\}) &= \mu(\{i,j\}) - \mu(\{i\}) - \mu(\{j\}) \\ m^\mu(\{i\}) &= \mu(\{i\}) \end{aligned}$$

By (10),

$$\begin{aligned} \phi^\mu(i) &= \mu(\{i\}) \\ &+ \frac{1}{2} \sum_{j \in N \setminus \{i\}} (\mu(\{i,j\}) - \mu(\{i\}) - \mu(\{j\})) \end{aligned} \quad (12)$$

$$I^\mu(\{i,j\}) = \mu(\{i,j\}) - \mu(\{i\}) - \mu(\{j\}) \quad (13)$$

4. Learning setting

We define in this section the necessary concepts to describe how to learn the GAI model.

4.1. Unknowns of the model

In order to learn all utility functions u_S , each attribute is discretized. For attribute $i \in N$, we keep only $\hat{X}_i \subseteq X_i$ with $|\hat{X}_i|$ finite in the learning phase. The unknowns of the GAI model are $\{u_S(z_S) : S \in \mathcal{S}, z_S \in \hat{X}_S\}$, where $\hat{X}_S = \prod_{i \in S} \hat{X}_i$. The elements of \hat{X}_i are denoted by $a_i^1, a_i^2, \dots, a_i^{p_i}$. In order to differentiate with the model u_S , we denote by

$$\hat{u} := \{\hat{u}_S(z_S) : S \in \mathcal{S}, z_S \in \hat{X}_S\} \quad (14)$$

the set of all unknowns. The number of unknowns is

$$M := \sum_{S \in \mathcal{S}} \prod_{i \in S} p_i. \quad (15)$$

4.2. Computation of the overall utility after elicitation

Once unknowns (14) are specified, let us determine how the utility u_S is computed for any $x_S \in X_S$.

Let us start with the simple case when S is a single attribute i . In order to deduce the value of u_i for all elements of X_i from $\hat{u}_i(a_i^1), \dots, \hat{u}_i(a_i^{p_i})$, we

assume that u_i is piecewise affine. If X_i is discrete, we can assume that this set is represented by integer numbers (the integer represents the Id of a label) so that interpolation still makes sense. Hence for $a_i^k \leq x_i \leq a_i^{k+1}$, we set

$$u_i(x_i) = \frac{x_i - a_i^k}{a_i^{k+1} - a_i^k} \hat{u}_i(a_i^{k+1}) + \frac{a_i^{k+1} - x_i}{a_i^{k+1} - a_i^k} \hat{u}_i(a_i^k). \quad (16)$$

When S contains more than one attribute, the idea is to perform multi-linear interpolation. Let

$$I = \{i \in N : x_i \notin \hat{X}_i\}.$$

For $i \in N$ we set

$$\begin{aligned} \underline{x}_i &= \operatorname{argmax}\{z_i \in \hat{X}_i : z_i \leq x_i\} \\ \bar{x}_i &= \operatorname{argmin}\{z_i \in \hat{X}_i : z_i \geq x_i\} \end{aligned}$$

Note that $\bar{x}_i = \underline{x}_i$ iff $i \in N \setminus I$. Then generalizing (16), we obtain

$$\begin{aligned} u_S(x_S) &= \sum_{A \subseteq I \cap S} \left[\prod_{i \in A} \frac{\bar{x}_i - x_i}{\bar{x}_i - \underline{x}_i} \times \prod_{i \in (I \cap S) \setminus A} \frac{x_i - \underline{x}_i}{\bar{x}_i - \underline{x}_i} \right. \\ &\quad \left. \times \hat{u}_S(\underline{x}_A, \bar{x}_{(I \cap S) \setminus A}, x_{S \setminus I}) \right], \end{aligned} \quad (17)$$

where $(\underline{x}_A, \bar{x}_{(I \cap S) \setminus A}, x_{S \setminus I})$ is an alternative that is equal to \underline{x}_k if $k \in A$, to \bar{x}_k if $k \in (I \cap S) \setminus A$, and to x_k if $k \in S \setminus I$.

Expression (17) can be put in the linear form

$$u_S(x_S) = \sum_{\hat{x}_S \in \hat{X}_S} \operatorname{coef}_{S, x_S}(\hat{x}_S) \hat{u}_S(\hat{x}_S) \quad (18)$$

where $\operatorname{coef}_{S, x_S}(\hat{x}_S)$ are non-negative coefficients.

4.3. Initial preferential information

The determination of the GAI model is carried out through a learning phase on the basis of some preferential information (learning data). These data are of the following type

- $x \succeq y$, where $x, y \in X$, means that x is judged at least as good as y ;
- $x \succeq \alpha$ (resp. $x \preceq \alpha$), where $x \in X$ and $\alpha \in \mathbb{R}$, means that the overall evaluation of x is at least (resp. at most) α .

The set of all preferential information provided by the decision maker is denoted by \mathcal{P} . Each piece of preferential information $P \in \mathcal{P}$ is turned into a linear constraint $T(P, \hat{u}) \geq 0$. The expression of $T(P, \hat{u})$ is :

- $T(P, \hat{u}) = U(x) - U(y)$ if P corresponds to datum " $x \succeq y$ ",
- $T(P, \hat{u}) = U(x) - \alpha$ (resp. $T(P, \hat{u}) = \alpha - U(x)$) if P corresponds to datum " $x \succeq \alpha$ " (resp. " $x \preceq \alpha$ "),

where U is related to \hat{u} by (2) and (17). In both cases, we write (using (18)),

$$T(P, \hat{u}) = \sum_{S \in \mathcal{S}} \sum_{\hat{x}_S \in \hat{X}_S} \text{coef}_P(S, \hat{x}_S) \hat{u}_S(\hat{x}_S) + c_P \quad (19)$$

where $\text{coef}_P(S, \hat{x}_S)$ are non-negative coefficients, and $c_P \in \mathbb{R}$.

5. Properties and conditions on u_S

From Section 4, we are looking for \hat{u} that fulfils the monotonicity conditions (3) and $T(P, \hat{u}) \geq 0$ for all $P \in \mathcal{P}$. First note that (3) can be rewritten as follows:

$$\begin{aligned} \forall x \in X \quad \forall i \in N \quad \forall y_i \in X_i \text{ with } y_i \succ_i x_i \\ U(y_i, x_{N \setminus \{i\}}) \geq U(x) \end{aligned} \quad (20)$$

where $(y_i, x_{N \setminus \{i\}})$ denotes the compound alternative that is equal to y_i on attribute i and to x_j on attributes for $j \in N \setminus \{i\}$. The number of elementary conditions contained in (20) is equal to

$$\sum_{i \in N} \left[(p_i - 1) \times \prod_{j \in N \setminus \{i\}} p_j \right]. \quad (21)$$

The following example shows that this number increases very fast and is not tractable with linear programming with a reasonable number of variables. Hence we are looking for a simpler monotonicity condition.

Example 1 Consider an example with $n = 10$ attributes and 5 unknowns per attribute ($p_i = 5$). Assume that \mathcal{S} is composed of all singletons and pairs of attributes. Then the overall number of unknowns is 1125 whereas the number of monotony constraints in (21) is 78 125 000.

5.1. Justification of assumptions under restrictive interaction

Without loss of generality, one can assume that $U(x) \geq 0$ for all $x \in X$ (provided that the α coefficients in the learning data are non-negative) and X is bounded.

In [4], the terms u_S can take both positive and negative signs. Let us start from the following example of a non-negative function $U(x_1, x_2)$ having a negative term:

$$U(x_1, x_2) = 2x_1 + x_2 - \max(x_1, x_2). \quad (22)$$

From the relation

$$\min(x_1, x_2) + \max(x_1, x_2) = x_1 + x_2,$$

(22) can be replaced by the equivalent expression:

$$U(x_1, x_2) = x_1 + \min(x_1, x_2). \quad (23)$$

In this illustrative example, the negative term has been replaced by a positive one. One wonders now if this process can be generalized to any function U . In other way, is it possible to transform any given GAI model in such a way that all terms become non-negative?

We focus here in a specific type of GAI model encompassing examples (22) and (23) – namely when interaction is allowed only among pairs of attributes. Before stating this result, let us first define the concept of 2-additive GAI model.

Definition 3 A overall assessment function U is said to be 2-additive if for every $i, j \in N$, $x_i, y_i \in X_i$, $x_j, y_j \in X_j$ and $z_{-i,j}, t_{-i,j} \in X_{-i,j}$

$$\begin{aligned} U(y_i, y_j, z_{-i,j}) - U(x_i, y_j, z_{-i,j}) - U(y_i, x_j, z_{-i,j}) \\ + U(x_i, x_j, z_{-i,j}) = U(y_i, y_j, t_{-i,j}) - U(x_i, y_j, t_{-i,j}) \\ - U(y_i, x_j, t_{-i,j}) + U(x_i, x_j, t_{-i,j}) \end{aligned}$$

The following results both provides a representation of 2-additive GAI models and shows that any 2-additive GAI model can be represented by only non-negative terms.

Proposition 4 U is 2-additive if and only if for every $t \in X$, there exists non-negative functions $U_{i,j} : X_i \times X_j \rightarrow \mathbb{R}_+$ (for every $\{i, j\} \subseteq N$) and $U_i : X_i \rightarrow \mathbb{R}_+$ (for every $i \in N$) such that for all $x \in X$ with $x_i \precsim_i t_i$ for all $i \in N$

$$U(x) = U(t) + \sum_{i=1}^n U_i(x_i) + \sum_{i,j} U_{i,j}(x_i, x_j). \quad (24)$$

Assume for simplicity that each attribute X_i is finite, $\precsim_i = \geq$ (the larger the value on the attribute the better), and a_i^1 is the smallest value of X_i , then taking $t = (a_1^1, \dots, a_n^1)$, expression (24) holds for all $x \in X$, with $U(t) \geq 0$.

5.2. Assumption on u_S

Generalizing Proposition 4, it suggests that if we have an expression of U with some negative terms u_S , one could find another expression where all u_S are non-negative. We also make a further assumption that each u_S appearing in U is monotonic, in order to reduce the overall number of monotonicity conditions. We make thus the following assumption.

Assumption 1 For every $S \in \mathcal{S}$ we assume that

- (i) $\forall x_S \in \hat{X}_S, u_S(x_S) \geq 0$,
- (ii) u_S is monotonic w.r.t. \precsim_i for all $i \in S$.

5.3. Monotonicity conditions

If X_i is an interval, we assume to have

$$a_i^1 < a_i^2 < \dots < a_i^{p_i}.$$

If X_i is a discrete set of labels, the ordering of the values $a_i^1, a_i^2, \dots, a_i^{p_i}$ is arbitrary (one can follow for instance the order in which the labels are stored in X_i).

Assumption 1 turns into the following monotonicity conditions: $\forall i \in N \ \forall S \in \mathcal{S}$ with $i \in S$, $\forall x_{S \setminus \{i\}} \in \widehat{X}_{S \setminus \{i\}} \ \forall k \in \{1, \dots, p_i - 1\}$,

$$a_i^k \succ_i a_i^{k+1} \Rightarrow \widehat{u}_S(a_i^k, x_{S \setminus \{i\}}) \geq \widehat{u}_S(a_i^{k+1}, x_{S \setminus \{i\}}) \quad (25)$$

$$a_i^k \prec_i a_i^{k+1} \Rightarrow \widehat{u}_S(a_i^k, x_{S \setminus \{i\}}) \leq \widehat{u}_S(a_i^{k+1}, x_{S \setminus \{i\}}) \quad (26)$$

$$a_i^k \equiv_i a_i^{k+1} \Rightarrow \widehat{u}_S(a_i^k, x_{S \setminus \{i\}}) = \widehat{u}_S(a_i^{k+1}, x_{S \setminus \{i\}}) \quad (27)$$

We denote by \mathcal{M}^{\geq} the set of 4-uplet $(S, i, k, x_{S \setminus \{i\}})$ such that (25) or (27) hold. We write these inequalities as

$$\sum_{z_S \in \widehat{X}_S} \text{coef}_{S,i,k,x_{S \setminus \{i\}}}(z_S) \widehat{u}_S(z_S) \geq 0. \quad (28)$$

we denote by $\mathcal{M}^=$ the set of 4-uplet $(S, i, k, x_{S \setminus \{i\}})$ such that (26) holds. We write these equalities as

$$\sum_{z_S \in \widehat{X}_S} \text{coef}_{S,i,k,x_{S \setminus \{i\}}}(z_S) \widehat{u}_S(z_S) = 0. \quad (29)$$

6. Elicitation of a GAI model

The elicitation problem amounts basically in finding $\widehat{u} \in \mathbb{R}^M$ (see (14) and (15)) such that $T(P, \widehat{u}) \geq 0$ for all $P \in \mathcal{P}$ and the monotonicity conditions (25), (26) and (27) are satisfied. This problem may not have solution. So, we first solve this problem.

6.1. Inconsistency check and analysis

6.1.1. Inconsistency repair

The set of constraints may be inconsistent in the sense that there is no solution to them. The solution consists in remove some constraints. Yet the user may enforce that some constraints in \mathcal{P} must be fulfilled. We denote by $\mathcal{P}^{\mathcal{F}} \subseteq \mathcal{P}$ the *flexible* constraints (those that can be relaxed) and by $\mathcal{P}^{\mathcal{NF}} = \mathcal{P} \setminus \mathcal{P}^{\mathcal{F}}$ the *non flexible* constraints (those that must be kept).

The standard way to solve a potential inconsistency is to introduce $\{0, 1\}$ slack variables on the flexible constraints. We introduce the following

MILP noted LP1:

$$\begin{aligned} & \text{Minimize} \quad \sum_{P \in \mathcal{P}^{\mathcal{F}}} \varepsilon_P \\ & \text{under} \quad \left| \begin{array}{l} \widehat{u} \in \mathbb{R}^M \\ (25), (26) \text{ and } (27) \\ \forall P \in \mathcal{P}^{\mathcal{F}} \quad \varepsilon_P \in \{0, 1\} \\ \forall P \in \mathcal{P}^{\mathcal{F}} \quad T(P, \widehat{u}) + M \varepsilon_P \geq 0 \\ \forall P \in \mathcal{P}^{\mathcal{NF}} \quad T(P, \widehat{u}) \geq 0 \end{array} \right. \end{aligned}$$

where M is a large number (larger than the largest possible value of U).

The set of constraints in \mathcal{P} is consistent if all values of ε are equal to 0: $\varepsilon_P = 0$ for all $P \in \mathcal{P}^{\mathcal{F}}$. In the alternative case, all the constraints for which ε is equal to 1 shall be removed or modified in order to recover consistency.

The user may not wish to modify all these constraints. Hence it is worth generating another possible set of constraints to be modified. Once LP1 is solved, we set $\lambda^{(1)} = \{P \in \mathcal{P}^{\mathcal{F}}, \varepsilon_P = 1\}$. Once we have found $\lambda^{(1)}, \dots, \lambda^{(q)}$, we find the next repair solution by solving the following program LP1($\lambda^{(1)}, \dots, \lambda^{(q)}$):

$$\begin{aligned} & \text{Minimize} \quad \sum_{P \in \mathcal{P}^{\mathcal{F}}} \varepsilon_P \\ & \text{under} \quad \left| \begin{array}{l} \widehat{u} \in \mathbb{R}^M \\ (25), (26) \text{ and } (27) \\ \forall P \in \mathcal{P}^{\mathcal{F}} \quad \varepsilon_P \in \{0, 1\} \\ \forall P \in \mathcal{P}^{\mathcal{F}} \quad T(P, \widehat{u}) + M \varepsilon_P \geq 0 \\ \forall P \in \mathcal{P}^{\mathcal{NF}} \quad T(P, \widehat{u}) \geq 0 \\ \forall k \in \{1, \dots, q\} \quad \sum_{P \in \mathcal{P}^{\mathcal{F}}, \lambda_P^{(k)}=1} (\varepsilon_P - 1) \leq 1 \end{array} \right. \end{aligned}$$

The last constraint ensures that the new solution ε is different from $\lambda^{(1)}, \dots, \lambda^{(q)}$.

The set of the q simplest repairs is generated by the following algorithm.

Algorithm 1 Function getRepairs():

```

If LP1 optimal functional is 0 then
  Return  $\emptyset$ ;
Else
   $\mathcal{R} \leftarrow \{\lambda^{(1)}\}$ , where  $\lambda^{(1)} = \{P \in \mathcal{P}^{\mathcal{F}}, \varepsilon_P = 1\}$ 
  and  $\varepsilon$  is solution to LP1;
  For  $k \in \{2, \dots, q\}$ 
    If LP1( $\mathcal{R}$ ) is infeasible then
      Return  $\mathcal{R}$ ;
    Else
       $\mathcal{R} \leftarrow \mathcal{R} \cup \{\lambda^{(k)}\}$ , where
       $\lambda^{(k)} = \{P \in \mathcal{P}^{\mathcal{F}}, \varepsilon_P = 1\}$  and
       $\varepsilon$  is solution to LP1( $\mathcal{R}$ );

```

At the end, the user shall choose one repair among $\lambda^{(1)}, \lambda^{(2)}, \dots$

Example 2 Let $N = \{1, 2, 3, 4\}$, $X_1 = X_2, X_3, X_4 = \{0, 1\}$, $\succ_1 = \succ_2 = \succ_3 = \succ_4 = \geq$ and

$$\mathcal{S} = \{\{1\}, \{2\}, \{3\}, \{4\}, \{1, 2\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{1, 3, 4\}\}.$$

Consider the following learning examples

- *Ex1*: $(1, 1, 1, 1) \geq 1$
- *Ex2*: $(0, 0, 0, 0) \leq 0$
- *Ex3*: $(0, 1, 1, 1) \leq 0$
- *Ex4*: $(1, 0, 1, 1) \leq 0$
- *Ex5*: $(1, 1, 0, 1) \leq 0$
- *Ex6*: $(1, 1, 1, 0) \leq 0$

The previous set of constraints is inconsistent with the GAI representation under structure \mathcal{S} . Running the previous algorithm, if the user only wants to see 5 alternative suggestions, we obtain the following repairs (ordered according to the level of complexity):

- Remove constraint *Ex1* $((1, 1, 1, 1) \geq 1)$;
- Remove constraint *Ex4* $((1, 0, 1, 1) \leq 0)$;
- Remove the two constraints *Ex3* $((0, 1, 1, 1) \leq 0)$ and *Ex5* $((1, 1, 0, 1) \leq 0)$;
- Remove the two constraints *Ex3* $((0, 1, 1, 1) \leq 0)$ and *Ex6* $((1, 1, 1, 0) \leq 0)$;
- Remove the two constraints *Ex5* $((1, 1, 0, 1) \leq 0)$ and *Ex6* $((1, 1, 1, 0) \leq 0)$.

6.1.2. Inconsistency explanation

From the algorithms described in the previous section, the DM has chosen a repair ε . Let $\mathcal{P}^C = \{P \in \mathcal{P}^F : \varepsilon_P = 0\}$ be the set of constraints in \mathcal{P}^F that can be kept. Hence constraints $\mathcal{P}^C \cup \mathcal{P}^{NF}$ can be kept (they are consistent together) and all constraints in $\mathcal{P}^F \setminus \mathcal{P}^C$ shall be changed or removed.

We now wish to explain why any constraint $P^* \in \mathcal{P}^F \setminus \mathcal{P}^C$ is inconsistent with $\mathcal{P}^C \cup \mathcal{P}^{NF}$ and (25), (26), (27). Finding an explanation of this inconsistency amounts to finding the smallest subset of $\mathcal{P}^C \cup \mathcal{P}^{NF}$ that is inconsistent with P^* . More precisely, we wish to find the smallest subset $Q \subseteq \mathcal{P}^C \cup \mathcal{P}^{NF}$ (w.r.t. set inclusion) such that the following set of constraints is inconsistent:

$$\begin{cases} \hat{u} \in \mathbb{R}^M \\ (25), (26) \text{ and } (27) \\ T(P^*, \hat{u}) \geq 0 \\ \forall P \in Q \quad T(P, \hat{u}) \geq 0 \end{cases} \quad (30)$$

Proposition 5 Let y be variables defined by $y_{S,i,k,z_{S \setminus \{i\}}}$ for $(S, i, k, z_{S \setminus \{i\}}) \in \mathcal{M}^{\geq} \cup \mathcal{M}^=$, y_P for all $P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \cup \{P^*\}$. Then Q is the set of constraints P in $\mathcal{P}^C \cup \mathcal{P}^{NF}$ s.t. $y_P > 0$, where one minimizes

$$\sum_{P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \cup \{P^*\}} \varepsilon_P$$

under the following constraints

$$\forall P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \quad y_P \geq 0 \quad (31)$$

$$y_{P^*} > 0 \quad (32)$$

$$\forall (S, i, k, z_{S \setminus \{i\}}) \in \mathcal{M}^{\geq} \quad y_{S,i,k,z_{S \setminus \{i\}}} \geq 0 \quad (33)$$

$$\forall P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \quad \varepsilon_P \in \{0, 1\} \text{ and } M y_P \geq \varepsilon_P \quad (34)$$

$$\forall S \in \mathcal{S} \quad \forall \hat{u}_S(\hat{x}_S)$$

$$\begin{aligned} & \sum_{P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \cup \{P^*\}} y_P \text{coef}_{S,x_S}(\hat{x}_S) \\ & + \sum_{(S,i,k,z_{S \setminus \{i\}}) \in \mathcal{M}^{\geq} \cup \mathcal{M}^=} y_{S,i,k,z_{S \setminus \{i\}}} \text{coef}_{S,i,k,z_{S \setminus \{i\}}}(\hat{x}_S) = 0 \end{aligned} \quad (35)$$

$$\sum_{P \in \mathcal{P}^C \cup \mathcal{P}^{NF} \cup \{P^*\}} y_P c_P < 0 \quad (36)$$

where M is a large number.

6.2. Minimizing the size of \mathcal{S}

Once we have remove inconsistency constraints, we can simply the set of coalitions in \mathcal{S} [4]. The idea is to remove all coalitions S in \mathcal{S} that are not necessary to represent the preferential information \mathcal{P} .

\mathcal{S} is composed of two sets: some of them are compulsory (e.g. for singletons), and the other ones shall be kept only when it is necessary. We write $\mathcal{S} = \mathcal{S}^F \cup \mathcal{S}^{NF}$, where superscript \mathcal{F} denotes the *flexible* coalitions (those that can be removed) and \mathcal{NF} denotes the *non flexible* coalitions (those that must be kept).

Let

$$\forall i \in \{1, \dots, n\} \quad \text{nb}c(i) = |\{S \in \mathcal{S}^F : |S| = i\}|.$$

We define the coefficient sequence $c(1), \dots, c(n)$ by

$$\begin{aligned} c(1) &= 0 \\ \forall i \in \{2, \dots, n\} \quad c(i) &= \sum_{j=1}^{i-1} c(j) \text{nb}c(j) + 1 \end{aligned}$$

We define the following linear program LP2:

$$\begin{aligned} & \text{Minimize} \quad \sum_{S \in \mathcal{S}^F} c(|S|) \varepsilon_S \\ & \text{under} \quad \begin{cases} \forall S \in \mathcal{S}^F \quad \varepsilon_S \in \{0, 1\} \\ \forall S \in \mathcal{S}^F \quad \forall x_S \in \hat{X}_S \quad \hat{u}_S(x_S) \leq M \varepsilon_S \\ (25), (26) \text{ and } (27) \\ \forall P \in \mathcal{P} \quad T(P, \hat{u}) \geq 0 \end{cases} \end{aligned}$$

Lemma 6 Minimizing $\sum_{S \in \mathcal{S}^F} c(|S|) \varepsilon_S$ in LP2 is equivalent to minimizing first the number of term of cardinality n , then minimizing the number of term of cardinality $n-1, \dots$, and finally minimizing the number of term of cardinality 2.

At the end, we replace \mathcal{S}^F by

$$\{S \in \mathcal{S}^F, \varepsilon_S = 1\}.$$

7. Interpretation of a GAI model

As we already discussed, in reference [4], the terms u_S can take both positive and negative signs, and the authors state that the sign of u_S (when S is a pair) corresponds to the sign of the interaction

among the attribute in S . Going back to our previous example where (22) and (23) are two equivalent expressions of the same function U . From the interpretation given in [4], the interaction between attributes 1 and 2 is negative in (22) and is positive in (23). Hence the sign of the interaction does not correspond to the sign of the terms in the GAI decomposition.

We rather generalize the approaches used in the context of the Choquet integral to interpret a capacity in terms of importance and interaction indices (see Section 3). First, we interpret the importance and interaction indices defined from a capacity to indices defined from the Choquet integral (see Section 7.1). Then, we define general importance and interaction indices for any GAI function. Finally, we provides the expressions for a 2-additive GAI model.

7.1. Importance and interaction indices for the Choquet integral

In the context of MCDA, the Shapley value can be seen as the mean importance of criteria and is thus a useful tool to interpret a capacity [1, 18]. The Shapley value is usually interpreted in the context of Cooperative Game Theory, that only from the capacity μ . This is completely satisfactory in the context of MCDA since it completely ignores the use of the Choquet integral.

The interpretation of the Shapley value (and the Shapley interaction indices) for the Choquet integral is basically due to J.L. Marichal who noticed that (see [19, proposition 5.3.3 page 141] and also [20, Definition 10.41 and Proposition 10.43 page 369])

$$I_S(\mu) = \int_{[0,1]^n} \Delta_S C_\mu(z) dz$$

where, for any function f , $\Delta_S f$ is defined recursively by

$$\begin{aligned} \Delta_S f(z) &= \Delta_i(\Delta_{S \setminus \{i\}} f)(x) \quad \text{for any } i \in S \\ \Delta_i f(z) &= f(z|z_i = 1) - f(z|z_i = 0) \end{aligned}$$

The Shapley value appears as the mean of relative amplitude of the range of C_μ w.r.t. criterion i , when the remaining variables take random values. What is true with Shapley value is also true for interaction indices.

The following relation is not difficult to prove:

$$I_S(\mu) = \int_{[0,1]^n} \frac{\partial^{|S|} C_\mu}{\partial z_S}(z) dz \quad (37)$$

where the partial derivative is piecewise continuous. Here the partial derivative is the local importance of C_μ at point z .

7.2. Importance and interaction indices for the GAI model

Firstly, we generalize expression (37) to function U . Compare to the case of an aggregation function from $[0, 1]^N$ onto $[0, 1]$, functions maps X onto \mathbb{R}_+ . Hence the integral shall be applied on X (assuming that the attribute correspond to intervals). More precisely, we restrict ourselves to the

$$V := [a_1^1, a_1^{p_1}] \times \cdots \times [a_n^1, a_n^{p_n}].$$

For the sake of simplicity, we assume that $\succsim_i = \geq$ for all attributes i . Generalizing (37), we define the interaction among criteria in S by

$$I_S(U) = \frac{1}{|V|} \int_V \frac{\partial^{|S|} U}{\partial z_S}(z) dz \quad (38)$$

7.3. Expression of the importance index for a 2-additive GAI model

We set $\phi_i(U) := I_{\{i\}}(U)$. By (24), U is 2-additive iff U takes the form

$$U(x) = C + \sum_{i=1}^n U_i(x_i) + \sum_{i,j} U_{i,j}(x_i, x_j) \quad (39)$$

where C is a constant (we take $t = (a_1^1, \dots, a_n^1)$ in Proposition 4). We only need to compute $\phi_i(U_k)$ and $\phi_i(U_{j,k})$.

Proposition 7 *We have*

$$\phi_i(U) = \phi_i(U_i) + \sum_{j \in N \setminus \{i\}} \phi_i(U_{i,j})$$

where

$$\phi_i(U_i) = \frac{1}{a_i^{p_i} - a_i^1} [\hat{u}_i(a_i^{p_i}) - \hat{u}_i(a_i^1)].$$

and

$$\begin{aligned} \phi_i(U_{i,j}) &= \frac{1}{a_j^{p_j} - a_j^1} \sum_{m'=1}^{p_j-1} \frac{a_j^{m'+1} - a_j^{m'}}{2} \\ &\quad \left[\frac{\hat{u}_{i,j}(a_i^{p_i}, a_j^{m'}) - \hat{u}_{i,j}(a_i^1, a_j^{m'})}{a_i^{p_i} - a_i^1} \right. \\ &\quad \left. + \frac{\hat{u}_{i,j}(a_i^{p_i}, a_j^{m'+1}) - \hat{u}_{i,j}(a_i^1, a_j^{m'+1})}{a_i^{p_i} - a_i^1} \right] \end{aligned}$$

7.4. Expression of the interaction index for a 2-additive GAI model

We set $I_{i,j}(U) := I_{\{i,j\}}(U)$.

Proposition 8 *We have*

$$I_{i,j}(U) = I_{i,j}(U_{i,j})$$

where

$$I_{i,j}(U_{i,j}) = \frac{1}{(a_i^{p_i} - a_i^1)(a_j^{p_j} - a_j^1)} \left[\widehat{u}_{i,j}(a_i^1, a_j^1) - \widehat{u}_{i,j}(a_i^1, a_j^{p_j}) - \widehat{u}_{i,j}(a_i^{p_i}, a_j^1) + \widehat{u}_{i,j}(a_i^{p_i}, a_j^{p_j}) \right]$$

8. Conclusion

This paper is devoted to the use of the GAI model in a MCDM context.

Firstly, the problem of which conditions are relevant to add on the GAI model to make it more easy to learn and interpret is addressed. From a user point of view, the partial utility functions u_S appearing in a GAI model are more easily interpreted if there are non-negative and monotonic. We consider a particular case of 2-additive GAI, where the definition is borrowed from the Choquet integral setting. In particular, we show that it is sufficient to consider only positive terms in the expression of a 2-additive GAI model.

Secondly, we are interested in the learning phase. As in references [4, 5], we interpret the learning examples provided by the user as constraints, which yields the use of linear programming to learn a GAI model. Our contribution is on handling inconsistent learning examples. We first generate several possible repairs (set of learning examples that need to be changed or removed to recover consistency). We also provide a new approach for explaining an inconsistency, based on the Farkas lemma.

Finally, we propose some indices to interpret a GAI model, based on a generalization of importance and interaction indices already at work for the Choquet integral. They give the expression of these indices for 2-additive GAI models.

References

- [1] M. Grabisch. The application of fuzzy integrals in multicriteria decision making. *European J. of Operational Research*, 89:445–456, 1996.
- [2] Ch. Gonzales and P. Perny. GAI networks for utility elicitation. In *Proceedings of the 9th International Conference on the Principles of Knowledge Representation and Reasoning*, pages 224–234, 2004.
- [3] Ch. Gonzales, P. Perny, and J.P. Dubus. Decision making with multiple objectives using GAI networks. *Artificial Intelligence Journal*, 175(7):1153–1179, 2000.
- [4] S. Greco, V. Mousseau, and R. Slowinski. Robust ordinal regression for value functions handling interacting criteria. *European Journal of Operational Research*, submitted.

- [5] D. Bigot, H. Fargier, J. Mengin, and B. Zanuttini. Using and learning GAI-decompositions for representing ordinal rankings. In *European Conference on Artificial Intelligence (ECAI) - Workshop on Preference Learning*, Montpellier, France, August 27–31 2012.
- [6] O.L. Mangasarian. *Nonlinear Programming*. McGraw-Hill Book Company, New York, 1969.
- [7] R. L. Keeney and H. Raiffa. *Decision with Multiple Objectives*. Wiley, New York, 1976.
- [8] G. Choquet. Theory of capacities. *Annales de l'Institut Fourier*, 5:131–295, 1953.
- [9] M. Grabisch and Ch. Labreuche. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. *Annals of Operation Research*, 175:247–286, 2010.
- [10] D. Bouyssou, M. Couceiro, C. Labreuche, J.-L. Marichal, and B. Mayag. Using choquet integral in machine learning: what can MCDA bring? In *Workshop from Multiple Criteria Decision Aid to Preference Learning*, Mons, Belgium, November 15–16 2012.
- [11] M. Sugeno. *Theory of fuzzy integrals and its applications*. PhD thesis, Tokyo Institute of Technology, 1974.
- [12] G. C. Rota. On the foundations of combinatorial theory I. Theory of Möbius functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 2:340–368, 1964.
- [13] K. Fujimoto, I. Kojadinovic, and J.-L. Marichal. Axiomatic characterizations of probabilistic and cardinal-probabilistic interaction indices. *Games and Economic Behavior*, 55:72–99, 2006.
- [14] L. S. Shapley. A value for n -person games. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games, Vol. II*, number 28 in Annals of Mathematics Studies, pages 307–317. Princeton University Press, 1953.
- [15] M. Grabisch. k -order additive discrete fuzzy measures and their representation. *Fuzzy Sets and Systems*, 92:167–189, 1997.
- [16] T. Murofushi and S. Soneda. Techniques for reading fuzzy measures (III): interaction index. In *9th Fuzzy System Symposium*, pages 693–696, Sapporo, Japan, May 1993.
- [17] M. Grabisch, J.-L. Marichal, and M. Roubens. Equivalent representations of set functions. *Mathematics of Operations Research*, 25(2):157–178, 2000.
- [18] M. Grabisch. Alternative representations of discrete fuzzy measures for decision making. *Int. J. of Uncertainty, Fuzziness, and Knowledge Based Systems*, 5:587–607, 1997.
- [19] J.-L. Marichal. *Aggregation operators for multicriteria decision aid*. PhD thesis, University of Liège, 1998.
- [20] M. Grabisch, J.L. Marichal, R. Mesiar, and E. Pap. *Aggregation functions*. Cambridge University Press, 2009.