GAI Networks for Utility Elicitation

Christophe Gonzales and **Patrice Perny**
LIP6 – pôle IA – University Paris 6
Paris, France
{Christophe.Gonzales,Patrice.Perny}@lip6.fr

Abstract

This paper deals with preference representation and elicitation in the context of multiattribute utility theory under risk. Assuming the decision maker behaves according to the EU model, we investigate the elicitation of generalized additively decomposable utility functions on a product set (GAI-decomposable utilities). We propose a general elicitation procedure based on a new graphical model called a GAI-network. The latter is used to represent and manage independences between attributes, as junction graphs model independences between random variables in Bayesian networks. It is used to design an elicitation questionnaire based on simple lotteries involving completely specified outcomes. Our elicitation procedure is convenient for any GAI-decomposable utility function, thus enhancing the possibilities offered by UCP-networks.

Keywords. Decision theory, graphical representations, preference elicitation, multiattribute expected utility, GAI-decomposable utilities.

Introduction

Over the last few years the growing interest in decision systems has stressed the need for compact representations of individual's beliefs and preferences, both for user-friendliness of elicitation and reduction of memory consumption. In Decision under Uncertainty, the diversity of individuals behaviors and application contexts have led to different mathematical models including Expected Utility (EU) (von Neumann & Morgenstern 1944; Savage 1954), Choquet EU (Schmeidler 1986), Qualitative EU (Dubois & Prade 1995), Generalized EU (Halpern & Chu 2003a; 2003b). The concern in compact numerical representations of preferences being rather recent, studies have mainly focused on EU and emphasized the potential of graphical models such as UCP-nets (Boutilier, Bacchus, & Brafman 2001) or influence diagrams (Howard & Matheson 1984; Shachter 1986).

Using EU requires both a numerical representation of the Decision Maker's (DM) preferences over all the possible outcomes (a *utility function*) and a family of probability distributions over these outcomes. In this paper we focus on the assessment of utility, which is usually performed through an interactive process. The DM is asked to answer "simple" questions such as "do you prefer a to b?" and a numerical representation follows.

Theoretically, the assessment of preferences over every pair of outcomes may be needed to elicit completely the DM's utility, but in practice the large size of the outcome set prevents such a procedure to be feasible. Fortunately, preferences often have an underlying structure that can be exploited to drastically reduce the elicitation burden. Several structures described in terms of different independence concepts have emerged from the multiattribute utility theory community (Keeney & Raiffa 1976; Fishburn 1970; Krantz *et al.* 1971) and led to different forms of utilities, the most popular of which being the additive and the multilinear decompositions. The particular independences both of these decompositions assume significantly simplify the elicitation procedures, yet as they compel the DM's preferences to satisfy very stringent constraints they are inadequate in many practical situations.

A "good" trade-off between easiness of elicitation and generality of the model can certainly be achieved by *Generalized Additive Independence* (GAI) (Fishburn 1970). This "weak" form of independence is sufficiently flexible to apply to most situations and as such deserves the elaboration of elicitation procedures. Although introduced in the sixties, GAI has not received many contributions yet. In particular, elicitation procedures suggested in the literature for GAI-decomposable utilities are not general purpose. They assume either that the utilities satisfy constraints imposed by CP-net structure (see UCP-nets (Boutilier, Bacchus, & Brafman 2001)) or that utilities are random variables (the prior distribution of which is known) and that the elicitation consists in finding an *a posteriori* utility distribution (Chajewska & Koller 2000; Chajewska, Koller, & Parr 2000). We feel that these additional assumptions might not be suitable in a significant number of practical decision problems. For instance, as we shall see later in this paper, there exist "simple" GAI-decomposable preferences that cannot be compacted by UCP-nets. Similarly, the existence of prior utility distributions is not always natural, for instance there is not much chance that a company manager facing a given decision problem may have a prior distribution of other managers utilities at hand. Hence an elicitation procedure applicable to any GAI decomposition should prove useful. The purpose of this paper is to propose such a procedure in the context of Decision Making under Risk. More precisely, we assume uncertainties are handled through probabilities and DM's preferences are consistent with EU.

The key idea in our elicitation procedure is to take advantage of a new graphical representation of GAI decompositions we call a GAI network. It is essentially similar to the junction graphs used for Bayesian networks (Shafer 1996; Jensen, Lauritzen, & Olesen 1990; 1990; Cowell *et al.* 1999). As such, it keep tracks of all the dependences between the different components of the utilities and the sequence of questions to be asked to the DM can be retrieved directly from this graph.

The paper is organized as follows: the first section provides necessary background in multiattribute utility theory. Then, a typical example showing how a GAI-decomposable utility can be elicited is presented. The third section introduces GAI networks, a graphical tool for representing GAI-decompositions. It also describes a general elicitation procedure relying on this network which applies to any GAI-decomposable utility, as well as a generic scheme for constructing the GAI network. We finally conclude by emphasizing some significant advantages of our elicitation procedure.

Utility Decompositions

In this paper, we address problems of decision making under risk (von Neumann & Morgenstern 1944) (or under uncertainty (Savage 1954)), that is the DM has a preference relation $\succsim_d$ over a set of decisions $\mathcal{D}$, "$d_1 \succsim_d d_2$" meaning the DM either prefers decision d_1 to d_2 or feels indifferent between both decisions. The consequence or *outcome* resulting from making a particular decision is uncertain and only known through a probability distribution over the set of all possible outcomes. Decisions can thus be described in terms of these distributions, i.e., to each decision is attached a *lottery*, that is a finite tuple of pairs (outcome, probability of the outcome), and to $\succsim_d$ is associated a preference relation $\succsim$ over the set of lotteries such that $d_1 \succsim_d d_2 \Leftrightarrow$ lottery$(d_1) \succsim$ lottery(d_2). Taking advantage of this equivalence, we will use lotteries instead of decisions in the remainder of the paper.

Let $\mathcal{X}$ be the finite set of outcomes and let $\mathcal{L}$ be the set of lotteries. $\langle p^1, x^1; p^2, x^2; \ldots; p^q, x^q\rangle$ denotes the lottery such that each outcome $x^i \in \mathcal{X}$ obtains with a probability $p^i > 0$ and $\sum_{i=1}^q p^q = 1$. Moreover, for convenience of notation, when unambiguous, we will note x instead of lottery $\langle 1, x\rangle$. Under some axioms expressing the "rational" behavior of the DM, (Savage 1954) and (von Neumann & Morgenstern 1944) have shown that there exist some functions $U : \mathcal{L} \mapsto \mathbb{R}$ and $u : \mathcal{X} \mapsto \mathbb{R}$, unique up to strictly positive affine transforms, such that $L_1 \succsim L_2 \Leftrightarrow U(L_1) \succsim U(L_2)$ for all $L_1, L_2 \in \mathcal{L}$ and $U(\langle p^1, x^1; \ldots; p^q, x^q\rangle) = \sum_{i=1}^q p^i u(x^i)$. Such functions assigning higher numbers to the preferred outcomes are called *utility functions*. As $U(\cdot)$ is the expected value of $u(\cdot)$, we say that the DM is an *expected utility maximizer*.

Eliciting $U(\cdot)$ consists in both assessing the probability distribution over the outcomes for each decision and eliciting function $u(\cdot)$. The former has been extensively addressed in the UAI community (Buntine 1994; Heckerman 1995). Now eliciting $u(\cdot)$ is in general a complex task as the size of $\mathcal{X}$ is usually very large. The first step to circumvent this problem is to remark that usually the set of outcomes can be described as a Cartesian product of *attributes* $\mathcal{X} = \prod_{i=1}^n X_i$, where each X_i is a finite set. For instance, a mayor facing the Decision Making problem of selecting one policy for the industrial development of his city can assimilate each policy to a lottery over outcomes defined as tuples of type (investment cost supported by the city, environmental consequence, impact on employment, etc). This particular structure can be exploited by observing that some independences hold between attributes. For instance, preferences over environment consequences should not depend on preferences over employment. Several types of independence have been suggested in the literature, taking into account different preference structures and leading to different functional forms of the utilities. The most usual is the following:

Definition 1 (Additive Independence) *Let L_1 and L_2 be any pair of lotteries and let p and q be their respective probability distributions over the outcome set. Then $X_1, \ldots, X_n$ are additively independent for $\succsim$ if p and q having the same marginals on every X_i implies that both lotteries are indifferent, i.e. $L_1 \succsim L_2$ and $L_2 \succsim L_1$ (or $L_1 \sim L_2$ for short).*

(Bacchus & Grove 1995) illustrates additive independence on the following example: let $\mathcal{X} = X_1 \times X_2$ where $X_1 = \{a_1, b_1\}$ and $X_2 = \{a_2, b_2\}$. Let L_1 and L_2 be lotteries whose respective probability distributions on $\mathcal{X}$ are p and q. Assume $p(a_1, a_2) = p(a_1, b_2) = p(b_1, a_2) = p(b_1, b_2) = 1/4$, $q(a_1, a_2) = q(b_1, b_2) = 1/2$ and $q(a_1, b_2) = q(b_1, a_2) = 0$. Then p and q have the same marginals on X_1 and X_2 since $p(a_1) = q(a_1) = 1/2$, $p(b_1) = q(b_1) = 1/2$, $p(a_2) = q(a_2) = 1/2$ and $p(b_2) = q(b_2) = 1/2$. So under additive independence, lotteries L_1 and L_2 should be indifferent.

As additive independence captures the fact that preferences only depend on the marginal probabilities on each attribute, it rules out interactions between attributes and thus results in the following simple form of utility (Bacchus & Grove 1995):

Proposition 1 *$X_1, \ldots, X_n$ are additively independent for $\succsim$ iff there exist some functions $u_i : X_i \mapsto \mathbb{R}$ such that $u(x) = \sum_{i=1}^n u_i(x_i)$ for any $x = (x_1, \ldots, x_n)$.*

Additive decomposition allows all u_i's to be elicited independently, thus considerably reducing the amount of questions required to determine $u(\cdot)$. However, as no interaction is possible among attributes, such functional form cannot be applied in many practical situations. Hence other types of independence have been introduced that capture more or less dependences. For instance *utility independence* of every attribute (Bacchus & Grove 1995) leads to a more general form of utility called *multilinear utility*:

$$u(x_1,\ldots,x_n)=\sum_{\emptyset\neq Y\subseteq\{1,\ldots,n\}} k_Y \prod_{i\in Y} u_i(x_i),$$

where the u_i's are scaled from 0 to 1. Multilinear utilities are more general than additive utilities but many interactions between attributes still cannot be taken into account by such functionals. Consider for instance the following example:

Example 1 Let $\mathcal{X}=X_1\times X_2$, where $X_1=\{$lamb, vegetable, beef$\}$ and $X_2=\{$red wine, white wine$\}$. Assume a DM has the following preferences over meals:

$$\begin{gathered}\text{(lamb, red wine)} \succ \text{(vegetable, red wine)}\\ \sim \text{(lamb, white wine)} \sim \text{(vegetable, white wine)}\\ \succ \text{(beef, red wine)} \succ \text{(beef, white wine)},\end{gathered}$$

that is the DM has some kind of lexicographic preference over food, and then some preference over wine. Then, if a multilinear utility $u(\text{food}, \text{wine}) = k_1u_1(\text{food}) + k_2u_2(\text{wine}) + k_3u_1(\text{food})u_2(\text{wine})$ existed, since utilities are scaled from 0 to 1, the above preference relations would imply that $u_1(\text{lamb}) = 1 \geq u_1(\text{vegetable}) = x \geq u_1(\text{beef}) = 0$ and that $u_2(\text{red wine}) = 1$ and $u_2(\text{white wine}) = 0$. But then the preference relations could be translated into a system of inequalities $k_1+k_2+k_3 > k_1x+k_2+k_3x = k_1 = k_1x > k_2 > 0$ having no solution, a contradiction. Consequently no multilinear utility can represent these DM preferences, although they are not irrational. ♦

Within multilinear utilities, interactions between attributes are taken into account using the products of subutilities on every attribute. The advantage is that the elicitation task remains reasonably tractable since only the assessments of the u_i's and of constants k_Y's are needed. But the price to pay is that many preference relations cannot be represented by such functions. One way out would be to keep the types of interactions between attributes unspecified, that is, separating the utility function into a sum of subutilities on sets of interacting attributes: this leads to the GAI decompositions. Those result from a generalization of additive utilities:

Definition 2 (Generalized Additive Independence) *Let L_1 and L_2 be any pair of lotteries and let p and q be their probability distributions over the outcome set. Let $Z_1,\ldots,Z_k$ be some subsets of $N=\{1,\ldots,n\}$ such that $N=\cup_{i=1}^k Z_i$ and let $X_{Z_i}=\{X_j : j\in Z_i\}$. Then $X_{Z_1},\ldots,X_{Z_k}$ are generalized additively independent for $\succsim$ if the equality of the marginals of p and q on all X_{Z_i}'s implies that $L_1\sim L_2$.*

As proved in (Bacchus & Grove 1995; Fishburn 1970) the following functional form of the utility called a GAI decomposition can be derived from generalized additive independence:

Proposition 2 *Let $Z_1,\ldots,Z_k$ be some subsets of $N=\{1,\ldots,n\}$ such that $N=\cup_{i=1}^k Z_i$. $X_{Z_1},\ldots,X_{Z_k}$ are generalized additively independent (GAI) for $\succsim$ iff there exist some real functions $u_i:\prod_{j\in Z_i}X_j\mapsto\mathbb{R}$ such that*

$$u(x)=\sum_{i=1}^k u_i(x_{Z_i}),\ \textit{for all } x=(x_1,\ldots,x_n)\in\mathcal{X},$$

where x_{Z_i} denotes the tuple of components of x having their index in Z_i.

Example 1 (continued) GAI decompositions allow great flexibility because they do not make any assumption on the kind of relations between attributes. Thus, if besides main course and wine, the DM wants to eat a dessert and a starter, her choice for the starter will certainly be dependent on that of the main course, but her preferences for desserts may not depend on the rest of the meal. This naturally leads to decomposing the utility over meals as $u_1(\text{starter}, \text{main course}) + u_2(\text{main course}, \text{wine}) + u_3(\text{dessert})$ and this utility corresponds precisely to a GAI decomposition. ♦

Note that the undecomposed utility $u(\cdot)$ and the additively decomposed utility $\sum_{i=1}^n u_i(\cdot)$ are special cases of GAI-decomposable utilities. The amount of questions required by the elicitation is thus closely related to the GAI decomposition itself. In practice, it is unreasonable to consider eliciting subutilities with more than 3 parameters. But GAI decompositions involving "small" X_{Z_i}'s can be exploited to keep the number of questions to a reasonable amount as shown in the next two sections.

Elicitation of a GAI-decomposable Utility

In this section, we will first present the general type of questions to be asked to the DM during the elicitation process and, then, we will specialize them to the GAI-decomposable model case.

Let $\succsim$ be a preference relation on the set $\mathcal{L}$ of all possible lotteries over an outcome set $\mathcal{X}$. Let x, y and z be three arbitrary outcomes such that the DM prefers making any decision the result of which is always outcome y (resp. x) to any decision resulting in x (resp. z), i.e., $y\succsim x\succsim z$. In terms of utilities, $u(y)\geq u(x)\geq u(z)$. Consequently, there exists a real number $p\in[0,1]$ such that $u(x)=pu(y)+(1-p)u(z)$, or equivalently, there exists a probability p such that $x\sim\langle p,y;1-p,z\rangle$. This gamble is illustrated on Figure 1. Knowing the values of

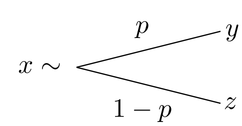


Figure 1: Gamble $x\sim\langle p,y;1-p,z\rangle$.

p, $u(y)$ and $u(z)$ thus completely determines that of $u(x)$. This is the very principle of utility elicitation under risk. In the remainder, to avoid testing which of the outcomes x, y or z are preferred to the others, for any three outcomes x^1,x^2,x^3, we will denote by $G(x^1,x^2,x^3)$ the gamble $x^{\sigma(2)}\sim\langle p,x^{\sigma(1)};1-p,x^{\sigma(3)}\rangle$ where σ is a permutation of $\{1,2,3\}$ such that $x^{\sigma(1)}\succsim x^{\sigma(2)}\succsim x^{\sigma(3)}$.

Assume that y and z correspond to the most and least preferred outcomes in $\mathcal{X}$ respectively, then all the x's in $\mathcal{X}$ are such that $y\succsim x\succsim z$, and the utility assigned to every outcome in $\mathcal{X}$ can be determined from the knowledge of p, $u(y)$ and $u(z)$. Moreover, as under von Neumann-Morgenstern's axioms utilities are unique up to strictly positive affine transforms, we can assume that $u(y) = 1$ and $u(z) = 0$. Hence there just remains to assess probabilities p. Different interactive procedures exist but they all share the same key idea: the DM is asked which of the following options she prefers: x or $\langle p, y; 1-p, z\rangle$ for a given value of p. If she prefers the first option, another similar question is asked with an increased value of p, else the value of p is decreased. When the DM feels indifferent between both options, p has been assessed.

Of course, as in practice $\mathcal{X}$ is a Cartesian product, $\mathcal{X}$'s size tends to increase exponentially with the number of attributes so that, as such, the above procedure cannot be completed using a reasonable number of questions. Fortunately, GAI decomposition helps reducing drastically the number of questions to be asked. The key idea can be illustrated with the following example:

Example 2 Consider an outcome set $\mathcal{X} = X_1 \times X_2 \times X_3$ and assume that $u(x_1, x_2, x_3) = u_1(x_1) + u_2(x_2, x_3)$. Then it is easily seen that gamble

$$(x_1, a_2, a_3) \sim \langle p, (y_1, a_2, a_3); 1-p, (z_1, a_2, a_3)\rangle$$

is equivalent to gamble

$$(x_1, b_2, b_3) \sim \langle p, (y_1, b_2, b_3); 1-p, (z_1, b_2, b_3)\rangle$$

as they both assert that $u_1(x_1) = pu_1(y_1) + (1-p)u_1(z_1)$. Hence, assuming preferences are stable over time, there is no need to ask the DM questions to determine the value of p in the second gamble: it is equal to that of p in the first one. Thus many questions can be avoided during the elicitation process. Note that in essence this property is closely related to a Ceteris Paribus statement (Boutilier *et al.* 1999). ♦

Now let us introduce our elicitation procedure with the following example:

Example 3 Let $\mathcal{X} = \prod_{i=1}^4 X_i$ and assume that utility $u : \mathcal{X} \mapsto \mathbb{R}$ over the outcomes is decomposable as $u(x_1, \ldots, x_4) = u_1(x_1, x_2) + u_2(x_2, x_3) + u_3(x_3, x_4)$. The elicitation algorithm consists in asking questions to determine successively the value of $u_1(\cdot)$, then that of $u_2(\cdot)$ and finally that of $u_3(\cdot)$.

Let (a_1, a_2, a_3, a_4) be an arbitrary outcome that will be used as a reference point. In the sequel, for notational convenience, instead of writing $x_{\{1,2\}}$ for (x_1, x_2) we shall write x_{12}. Let us show that we may assume without loss of generality that:

$$\begin{array}{ll} u_1(b_1, a_2) = 1, & u_1(a_1, x_2) = 0 \text{ for all } x_2 \in X_2, \\ u_3(a_3, a_4) = 0, & u_2(a_2, x_3) = 0 \text{ for all } x_3 \in X_3. \end{array} \quad (1)$$

Assume the DM's preferences are representable by a utility

$$v(x_1, \ldots, x_4) = v_1(x_1, x_2) + v_2(x_2, x_3) + v_3(x_3, x_4)$$

on the outcome set such that $v(\cdot)$ does not necessarily satisfy Eq. (1). Let

$$u_1(x_1, x_2) = v_1(x_1, x_2) - v_1(a_1, x_2).$$

Then $v(x_1, \ldots, x_4) = u_1(x_1, x_2) + [v_2(x_2, x_3) + v_1(a_1, x_2)] + v_3(x_3, x_4)$ and $v_2(x_2, x_3) + v_1(a_1, x_2)$ is a function on $X_2 \times X_3$ and $u_1(a_1, x_2) = 0$ for all x_2's. It can thus be said that $v_2(\cdot)$ has "*absorbed*" a part of $v_1(\cdot)$. Similarly, some part of $v_2(\cdot)$ may be absorbed by $v_3(\cdot)$ in such a way that the resulting $u_2(a_2, x_3) = 0$ for all x_3's: it is sufficient to define

$$\begin{aligned} u_2(x_2, x_3) = \ & v_2(x_2, x_3) + v_1(a_1, x_2) \\ & - v_2(a_2, x_3) - v_1(a_1, a_2). \end{aligned}$$

$v(x_1, \ldots, x_4)$ thus equals to $u_1(x_1, x_2) + u_2(x_2, x_3) + v_3(x_3, x_4) + v_2(a_2, x_3) + v_1(a_1, a_2)$. Note that $u_3(x_3, x_4) = v_3(x_3, x_4) + v_2(a_2, x_3) + v_1(a_1, a_2)$ is a function over $X_3 \times X_4$ as $v_1(a_1, a_2)$ is a constant.

Von Neumann-Morgenstern's utilities being unique up to positive affine transforms, it can be assumed without loss of generality that $u(a_1, a_2, a_3, a_4) = 0$ and that $u(b_1, a_2, a_3, a_4) = 1$ for some arbitrary $b_1 \in X_1$ such that outcome $(b_1, a_{234}) \succsim (a_1, a_{234})$, hence resluting in $u_3(a_3, a_4) = 0$ and $u_1(b_1, a_2) = 1$. Consequently, hypotheses (1) may be assumed without loss of generality.

Thus, the assessment of $u_1(x_1, a_2)$ for all x_1's can be derived directly from gambles such as:

$$\begin{array}{c} (x_1, a_{234}) \sim \langle p, (b_1, a_{234}); 1-p, (a_1, a_{234})\rangle \\ \text{also denoted as } G((b_1, a_{234}), (x_1, a_{234}), (a_1, a_{234})), \end{array} \quad (2)$$

as they are equivalent to $u_1(x_1, a_2) = p$. Note that in the above gambles lotteries only differ by the first attribute value, hence the questions asked to the DM should not be cognitively too complicated and the DM should not have difficulties answering them. Then

$$G((b_1, a_2, a_{34}), (a_1, x_2, a_{34}), (a_1, a_2, a_{34})) \quad (3)$$

determines the value of $u_2(x_2, a_3)$. For instance, if $(b_1, a_2, a_{34}) \succ (a_1, x_2, a_{34})$, then the above gamble is equivalent to:

$$(a_1, x_2, a_{34}) \sim \langle q, (b_1, a_2, a_{34}); 1-q, (a_1, a_2, a_{34})\rangle,$$

which implies that $u_2(x_2, a_3) = q$. Combining Eq. (3) with

$$G((b_1, a_2, a_{34}), (x_1', x_2, a_{34}), (a_1, a_2, a_{34})), \quad (4)$$

where x_1' is an arbitrary value of X_1, the determination of $u_1(x_1', x_2)$ follows. Note that until now all calls to function $G(\cdot)$, and especially in equations (3) and (4), shared the same first and third outcomes, i.e., (b_1, a_2, a_{34}) and (a_1, a_2, a_{34}). Note also that the gambles remain cognitively "simple" as most of the attributes are the same for all outcomes. Now, the value of $u_1(x_1', x_2)$ is sufficient to induce from $G((x_1', x_2), (x_1, x_2), (a_1, x_2))$ the values of all the $u_1(x_1, x_2)$'s and the determination of $u_1(\cdot)$ is completed.

The same process applies to assess $u_2(\cdot)$. First, using gambles similar to that of Eq. (3), i.e., $G((b_1, a_2, a_{34}), (a_1, b_2, a_{34}), (a_1, a_2, a_{34}))$, $u_2(b_2, a_3)$ can be assessed for arbitrary values b_2 of X_2. Then $G((a_1, b_2, a_{34}), (a_1, x_2, a_{34}), (a_1, a_2, a_{34}))$ will enable the determination of the $u_2(x_2, a_3)$'s for all x_2's (in fact, they will involve terms in $u_1(\cdot)$ and $u_2(\cdot)$ but as $u_1(\cdot)$ has been elicited, only the $u_2(\cdot)$'s remain unknown). Once the $u_2(x_2, a_3)$'s are known, gambles similar to those of Eq. (3) and Eq. (4) but applied to X_2, X_3 instead of X_1, X_2 lead to the complete determination of $u_2(\cdot)$.

Finally as function $u_3(\cdot)$ is the only remaining unknown, $u_3(x_3, x_4)$ can be elicited directly using any gamble involving two "elicited" outcomes. For instance $G((b_1, a_{23}, a_4), (a_1, a_{23}, x_4), (a_1, a_{23}, a_4))$ will determine the $u_3(a_3, x_4)$'s for all values of x_4 and, then, $G((a_{12}, a_3, b_4), (a_{12}, x_3, x_4), (a_{12}, a_3, a_4))$ will complete the assessment of $u_3(\cdot)$. ♦

Note that only a few attributes differed in the outcomes of each of the above gambles, hence resulting in cognitively simple questions. At first sight, this elicitation scheme seems to be a *ad hoc* procedure but, as we shall see in the next section, it proves to be in fact quite general.

GAI Networks

To derive a general scheme from the above example, we introduce a graphical structure we call a *GAI network*, which is essentially similar to the junction graphs used in Bayesian networks (Jensen 1996; Cowell *et al.* 1999):

Definition 3 (GAI network) *Let $Z_1, \ldots, Z_k$ be some subsets of $N = \{1, \ldots, n\}$ such that $\bigcup_{i=1}^k Z_i = N$. Assume that $\succsim$ is representable by a GAI-decomposable utility $u(x) = \sum_{i=1}^k u_i(x_{Z_i})$ for all $x \in \mathcal{X}$. Then a GAI network representing $u(\cdot)$ is an undirected graph $G = (V, E)$, satisfying the following properties:*

1. *$V = \{X_{Z_1}, \ldots, X_{Z_k}\}$;*
2. *For every $(X_{Z_i}, X_{Z_j}) \in E$, $Z_i \cap Z_j \neq \emptyset$. Moreover, for every pair of nodes X_{Z_i}, X_{Z_j} such that $Z_i \cap Z_j = T_{ij} \neq \emptyset$, there exists a path in G linking X_{Z_i} and X_{Z_j} such that all of its nodes contain all the indices of T_{ij} (Running intersection property).*

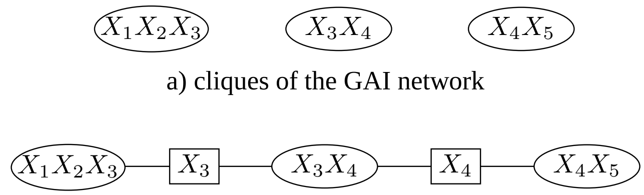
Nodes of V are called cliques. *Moreover, every edge $(X_{Z_i}, X_{Z_j}) \in E$ is labeled by $X_{T_{ij}} = X_{Z_i \cap Z_j}$, which is called a* separator.

Throughout this paper, cliques will be drawn as ellipses and separators as rectangles. The rest of this section will be devoted to the construction of GAI networks, and especially GAI trees, from GAI decompositions of utilities, and an elicitation procedure applicable to any GAI tree will be inferred from the example of the preceding section.

From GAI Decompositions to GAI Networks

For any GAI decomposition, Definition 3 is explicit as to which cliques should be created: these are simply the sets of variables of each subutility. For instance, if $u(x_1, \ldots, x_5) = u_1(x_1, x_2, x_3) + u_2(x_3, x_4) + u_3(x_4, x_5)$ then, as shown in Figure 2.a, cliques are $\{X_1, X_2, X_3\}$, $\{X_3, X_4\}$ and $\{X_4, X_5\}$.

Property 2 of Definition 3 gives us a clue for determining the set of edges of a GAI network: the algorithm constructing this set should always preserve the running intersection property. A simple — although not always efficient — way to construct the edges thus simply consists in linking cliques that have some nodes in common. Hence the following algorithm:



a) cliques of the GAI network

b) edges of the GAI network

Figure 2: The construction of a GAI network.

Algorithm 1 (Construction of a GAI network)
construct set $V = \{X_{Z_1}, \ldots, X_{Z_k}\}$;
for $i \in \{1 \ldots, k-1\}$ **do**
  for $j \in \{i+1 \ldots, k\}$ **do**
    if $Z_i \cap Z_j \neq \emptyset$ **then**
      add edge (X_{Z_i}, X_{Z_j}) to E
    fi
  done
done

Applying this algorithm on set $V = \{\{X_1, X_2, X_3\}, \{X_3, X_4\}, \{X_4, X_5\}\}$, sets $\{X_1, X_2, X_3\}$ and $\{X_3, X_4\}$ having a nonempty intersection, an edge should be created between these two cliques. Similarly, edge $(\{X_3, X_4\}, \{X_4, X_5\})$ should also be added as X_4 belongs to both cliques. Consequently the network of Figure 2.b is a GAI network representing $u(x_1, \ldots, x_5) = u_1(x_1, x_2, x_3) + u_2(x_3, x_4) + u_3(x_4, x_5)$.

As we shall see in the next subsection, GAI trees are more suitable than multiply-connected networks for conducting the elicitation process. Unfortunately, GAI networks representing utility decompositions often contain cycles. For instance, consider the following decomposition: $u(x_1, x_2, x_3, x_4) = u_1(x_1, x_2) + u_2(x_2, x_3) + u_3(x_3, x_4) + u_4(x_4, x_1)$. Then the only possible GAI network is that of Figure 3.

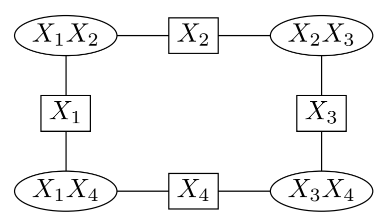
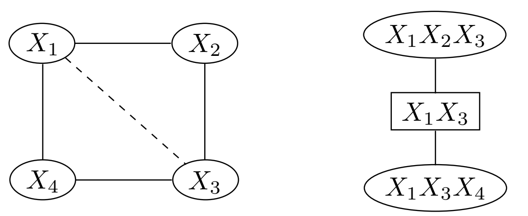


Figure 3: A GAI network containing a cycle.

Unlike GAI trees where a sequence of questions revealing the DM's utility function naturally arises, GAI multiply-connected networks do not seem to be appropriate to easily infer the sequence of questions to ask to the DM. Fortunately, they can be converted into GAI trees using the same triangulation techniques as in Bayesian networks (Kjærulff 1990; Darwiche & Hopkins 2001):

Algorithm 2 (Construction of a GAI tree)
1/ create a graph $G' = (V', E')$ such that
a/ $V' = \{X_1, \ldots, X_n\}$;
b/ edge (X_i, X_j) belongs to E' iff there exists a subutility containing both X_i and X_j
2/ triangulate G'
3/ derive from the triangulated graph a junction tree: the GAI tree

For instance, consider again the GAI network of Figure 3 representing utility $u(x_1, x_2, x_3, x_4) = u_1(x_1, x_2) + u_2(x_2, x_3) + u_3(x_3, x_4) + u_4(x_4, x_1)$. Graph G' constructed on step 1 of the above algorithm is depicted on Figure 4.a: the nodes of this graph are X_1, X_2, X_3, X_4, i.e., they correspond to the attributes of the utility. As function $u_1(\cdot)$ is defined over $X_1 \times X_2$, G' contains edge (X_1, X_2). Similarly, functions $u_2(\cdot)$, $u_3(\cdot)$ and $u_4(\cdot)$ imply that E' contains edges (X_2, X_3), (X_3, X_4) and (X_4, X_1), hence resulting in the solid edges in Figure 4.a. Note that graph G' corresponds to a CA-independence map of (Bacchus & Grove 1995).



a. attribute dependences   b. final GAI tree

Figure 4: From a GAI network to a GAI tree.

On step 2, G' is triangulated using any triangulation algorithm (Becker & Geiger 2001; Kjærulff 1990; Olesen & Madsen 2002), for instance using the following one:

Algorithm 3 (triangulation) *Let $G' = (V', E')$ be an undirected graph, where $V' = \{X_1, \ldots, X_n\}$. Let $adj(X_i)$ denote the set of nodes adjacent to X_i in G'. A node $X_i \in V'$ is said to be eliminated from graph G' when*

i) the edges $(adj(X_i) \times adj(X_i)) \backslash E'$ are added to E' so that $adj(X_i) \cup \{X_i\}$ becomes a clique;

ii) the edges between X_i and its neighbors are removed from E', as well as X_i from V'.

Let σ be any permutation of $\{1, \ldots, n\}$. Let us eliminate $X_{\sigma(1)}, X_{\sigma(2)}, \ldots, X_{\sigma(n)}$ successively and call E'_T the set of edges added to graph G' by these eliminations. Then graph $G'_T = (V', E' \cup E'_T)$ is triangulated.

This triangulation algorithm, when applied with elimination sequence X_2, X_3, X_1, X_4, precisely produces the graph of Figure 4.a, in which edges in E'_T are drawn with dashed lines.

Step 3 consists in constructing a new graph the nodes of which are the cliques of G' (i.e., maximal complete subgraphs of G'): here, $\{X_1, X_2, X_3\}$ and $\{X_1, X_3, X_4\}$ (see Figure 4.a). The edges between these cliques derive from the triangulation (Cowell *et al.* 1999; Kjærulff 1990; Rose 1970): each time a node X_i is eliminated, it will either create a new clique C_i or a subclique of an already existing clique C_i. In both cases, associate C_i to each X_i. Once a node X_i is eliminated, it cannot appear in the cliques created afterward. However, just after X_i's elimination, all the nodes in $C_i \backslash \{X_i\}$ still form a clique, hence the clique associated to the first eliminated node in $C_i \backslash \{X_i\}$ contains $C_i \backslash \{X_i\}$. Thus linking C_i to this clique ensures the running intersection property. In our example, clique $\{X_1, X_2, X_3\}$ is associated to node X_2 while clique $\{X_1, X_3, X_4\}$ is associated to the other nodes. As X_2 is the first eliminated node, we shall examine clique $\{X_1, X_2, X_3\}$. $C_i \backslash \{X_i\}$ is thus equal to $\{X_1, X_3\}$. Among these nodes, X_3 is the first to be eliminated and clique $\{X_1, X_3, X_4\}$ is associated to this node. Hence, there should exist an edge between cliques $\{X_1, X_2, X_3\}$ and $\{X_1, X_3, X_4\}$. As each clique is linked to at most one other clique, the process ensures that the resulting graph is actually a tree (see Figure 4.b).

Note that the GAI tree simply corresponds to a coarser GAI decomposition of the DM's utility function, i.e., it simply occults some known local independences, but this is the price to pay to make the elicitation process easy to perform.

Utility Elicitation in GAI Trees

This subsection first translates into a GAI tree-conducted algorithm the elicitation process of the preceding section and, then, a general algorithm is derived.

Example 3 (continued) The GAI network related to Example 3 is shown on Figure 5: ellipses represent the attributes of each subutility and rectangles the intersections between pairs of ellipses. Separators are essential for elicitation because they capture all the dependencies between sets of attributes. For instance separator X_2 reveals that clique X_1X_2 is independent of the rest of the graph for any fixed value of X_2. Hence answers to questions involving gambles on outcomes of type $(\cdot, a_2, a_3, a_4)$ do not depend on a_3, a_4, thus simplifying the elicitation of $u_1(\cdot, a_2)$.

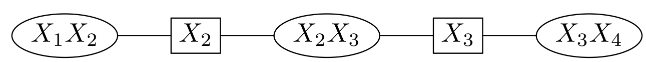


Figure 5: The GAI tree of Example 3.

The elicitation process described in Example 3 can be reformulated using the GAI tree as follows: we started with an outer clique, i.e., a clique connected to at most one separator. The clique we chose was X_1X_2. Function $u_1(\cdot)$ was assessed for every value of the attributes in the clique except those in the separator (here X_2) that were kept to the reference point a_2. This led to assessing $u_1(x_1, a_2)$ for all x_1's using Eq. (2)'s gamble:

$$G((b_1, a_{234}), (x_1, a_{234}), (a_1, a_{234})).$$

Then the values of the attributes in the separator were changed to, say x_2, and $u_1(\cdot)$ was elicited for every value of the attributes in clique X_1X_2 except those in the separator that were kept to x_2. This was performed using the gambles of Eq. (3) and (4), as well as gambles similar to the one above, i.e.,

$$G((b_1,a_2,a_{34}),(a_1,x_2,a_{34}),(a_1,a_2,a_{34})),$$
$$G((b_1,a_2,a_{34}),(x'_1,x_2,a_{34}),(a_1,a_2,a_{34})),$$
$$G((x'_1,x_2,a_{34}),(x_1,x_2,a_{34}),(a_1,x_2,a_{34})).$$

After $u_1(\cdot)$ was completely determined, clique X_1X_2 and its adjacent separator were removed from the network and we applied the same process with another outer clique, namely clique X_2X_3: using gamble $G((b_1,a_2,a_{34}),(a_1,b_2,a_{34}),(a_1,a_2,a_{34}))$, $u_2(b_2,a_3)$ could be determined. Then gamble

$$G((a_1,b_2,a_3,a_4),(a_1,x_2,a_3,a_4),(a_1,a_2,a_3,a_4))$$

was used to assess the value of $u_2(x_2,a_3)$ for any x_2 in X_2. In other words, we assessed the value of $u_2(\cdot)$ for every value of the attributes in the clique except those in the separator (X_3) that were kept to the reference point a_3. Once the $u_2(x_2,a_3)$'s were known, $u_2(\cdot)$ was determined for different values of x_3 using gambles

$$G((b_1,a_2,a_3,a_4),(a_1,a_2,x_3,a_4),(a_1,a_2,a_3,a_4)),$$
$$G((b_1,a_2,a_3,a_4),(a_1,b_2,x_3,a_4),(a_1,a_2,a_3,a_4)),$$
$$G((a_1,b_2,x_3,a_4),(a_1,x_2,x_3,a_4),(a_1,a_2,x_3,a_4)),$$

i.e., the values of the attributes in the separator were changed to x_3 and $u_2(\cdot)$ was elicited for every value of the attributes in clique X_2X_3 except those in the separator that were kept to x_3, and so on.

All cliques can thus be removed by induction until there remains only one clique. This one deserves a special treatment as the hypotheses of Eq. (1) specifying that, when we elicit a subutility $u_i(\cdot)$, $u_i(\cdot)=0$ whenever the value of the attributes not in the separator equal those of the reference point, apply to every clique except the last one. When determining the value of the utility of the last clique, all the other subutilities are known and a direct elicitation can thus be applied. ♦

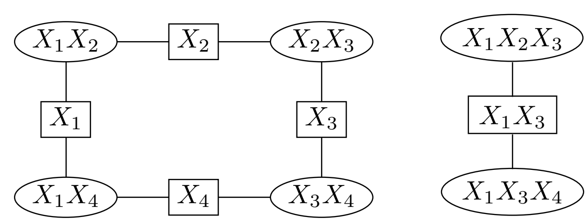
The above example suggests the following general elicitation procedure, which is applicable to any GAI tree: let $\succsim$ be a preference relation on lotteries over the outcome set $\mathcal{X}$. Let $Z_1,\dots,Z_k$ be some subsets of $N=\{1,\dots,n\}$ such that $N=\cup_{i=1}^k Z_i$ and such that $u(x)=\sum_{i=1}^k u_i(x_{Z_i})$ is a GAI-decomposable utility. Assume that the X_{Z_i}'s are such that they form a GAI tree $G=(V,E)$ and that for every i, once all X_{Z_j}'s, $j<i$, have been removed from G as well as their adjacent edges and separators, X_{Z_i} has only one adjacent separator left we will denote by X_{S_i}. In other words, the X_{Z_i}'s are ordered giving priorities to outer nodes. Call $C_i=Z_i\backslash S_i$, and let $C_k=Z_k\backslash S_{k-1}$. Let $(a_1,\dots,a_n)$ and $(b_1,\dots,b_n)$ be arbitrary outcomes of $\mathcal{X}$ such that $(b_{C_i},a_{N\backslash C_i})\succ(a_{C_i},a_{N\backslash C_i})$ for all i's. Then algorithm 4 completely determines the value of each subutility which can then be stored in cliques, thus turning the GAI network into a compact representation of $u(\cdot)$.

Of course, algorithm 4 can be applied whichever way the GAI tree is obtained. In particular, it can be applied on GAI trees resulting from triangulations. For the latter, the algorithm may be improved taking into account the knowledge of the GAI decomposition before triangulation.

Algorithm 4

$u_1(b_{C_1},a_{N\backslash C_1})\leftarrow 1$; $u_1(a_N)\leftarrow 0$
for all i *in* $\{1,\dots,k\}$ **and all** x_{S_i} **do**
    $u_i(a_{C_i},x_{N\backslash C_i})\leftarrow 0$
done
for all i *in* $\{1,\dots,k-1\}$ **do**
    if $i\neq 1$ **then**
        compute $u_i(b_{C_i},a_{N\backslash C_i})$ *using*
            $G((b_{C_1},a_{N\backslash C_1}),(b_{C_i},a_{N\backslash C_i}),(a_N))$
    endif
    for all x_{S_i} **do**
        if $x_{S_i}\neq a_{S_i}$ **then**
            compute $u_i(b_{C_i},x_{S_i},a_{N\backslash Z_i})$ *using*
                $G((b_{C_1},a_{N\backslash C_1}),(x_{S_i},a_{N\backslash S_i}),(a_N))$
                and $G((b_{C_1},a_{N\backslash C_1}),(b_{C_i},x_{S_i},a_{N\backslash Z_i}),(a_N))$
        endif
        for all x_{Z_i} **do**
            compute $u_i(x_{Z_i})$ *using* $G((b_{C_i},x_{S_i},a_{N\backslash Z_i}),$
                $(x_{Z_i},a_{N\backslash Z_i}),(a_{C_i},x_{S_i},a_{N\backslash Z_i}))$
        done
    done
done
/* *computation of the final clique* */
compute $u_k(b_{C_k},a_{S_{k-1}})$ *using*
    $G((b_{C_1},a_{N\backslash C_1}),(b_{C_k},a_{N\backslash C_k}),(a_N))$
for all x_{Z_k} **do**
    compute $u_k(x_{Z_k})$ *using*
        $G((b_{C_k},a_{N\backslash C_k}),(x_{Z_k},a_{N\backslash Z_k}),(a_N))$
done

Consider for instance the following GAI decomposition: $u(x_1,x_2,x_3,x_4)=u_1(x_1,x_2)+u_2(x_2,x_3)+u_3(x_3,x_4)+u_4(x_4,x_1)$, representable by the GAI network of Figure 6.a and inducing the GAI tree of Figure 6.b, or equivalently the GAI decomposition $u(x_1,\dots,x_4)=v_1(x_1,x_2,x_3)+v_2(x_1,x_3,x_4)$. Applying directly the elicitation process in



a. original GAI network   b. final GAI tree

Figure 6: A GAI tree resulting from a triangulation.

the graph of Figure 6.b would be quite inefficient as many questions would be asked to the DM although their answers could be computed from the answers given to previous questions. For instance, assume that X_1 (resp. X_2; X_3) can take values a_1,b_1 (resp. a_2,b_2; a_3,b_3). Then, as obviously $v_1(x_1,x_2,x_3)=u_1(x_1,x_2)+u_2(x_2,x_3)$, the above elicitation algorithm ensures that

$$v_1(a_1,a_2,a_3)=u_1(a_1,a_2)+u_2(a_2,a_3)=0.$$

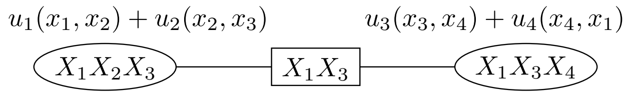


Figure 7: Subutilities in a GAI tree.

But, then,

$$\begin{aligned} v_1(b_1,a_2,b_3) &= u_1(b_1,a_2)+u_2(a_2,b_3) \\ &= u_1(b_1,a_2)+u_2(a_2,a_3)+ \\ &\quad u_1(a_1,a_2)+u_2(a_2,b_3) \\ &= v_1(b_1,a_2,a_3)+v_1(a_1,a_2,b_3). \end{aligned}$$

Hence, after the elicitation of both $v_1(b_1,a_2,a_3)$ and $v_1(a_1,a_2,b_3)$, that of $v_1(b_1,a_2,b_3)$ can be dispensed with. Intuitively, such questions can be found simply by setting down the system of equations linking the v_i's to the u_i's and identifying colinear vectors.

In GAI trees, the running intersection property ensures that the questions related to the subutilities of outer cliques are sufficient to determine unambiguously these subutilities. When the GAI networks are multiply-connected, this property does not hold anymore: the equations resulting from questions do often involve several unknown subutility values. Consequently, in such networks, questions are used to fill a system of linear equations on subutility values and, when the elicitation process is completed, this system is solved, thus producing values for all subutilities. GAI multiply-connected networks are thus less user-friendly than GAI trees to perform the elicitation process. Moreover, as the subutility values remain unknown until the elicitation process is completed, determining the next question to ask is less obvious than in GAI trees because we must find a question that will not add an equation colinear with the rest of the linear system, hence this requires additional tests.

Conclusion

In this paper, we provided a general algorithm for eliciting GAI-decomposable utilities. Unlike UCP-nets, GAI networks do not assume some CP-net structure and thus extend the range of application of GAI-decomposable utilities. For instance, consider a DM having some preferences over some meals constituted by a main course (either a stew or some fish), some wine (red or white) and a dessert (pudding or an ice cream), in particular

$$\begin{aligned} &\text{(stew,red wine,dessert)} \succ \text{(fish,white wine,dessert)} \\ \succ\ &\text{(stew,white wine,dessert)} \succ \text{(fish,red wine,dessert)} \end{aligned}$$

for any dessert. Moreover, assume that the DM would like to suit the wine to the main course and she prefers having ice cream when she eats a stew. Then such preferences can be represented efficiently by $u(\text{meal}) = u_1(\text{course, wine}) + u_2(\text{course, dessert})$ and thus be compacted by the associated GAI network. Nevertheless, since preferences over courses depend on wine and conversely, and since there exists some dependence between courses and desserts, UCP-nets do not help in compacting utility function $u(\cdot)$ despite its GAI decomposability.

Another specificity of our procedure is that we always consider gambles over completely specified outcomes, i.e., including all the attributes. This is an advantage because answers to questions involving only a subset of attributes are not easily interpretable. Consider for instance a multi-attribute decision problem where the multi-attribute space is $\mathcal{X} = X_1 \times X_2 \times X_3 \times X_4$, with $X_1 = \{a_1, c_1, b_1\}$, $X_2 = \{a_2, c_2, b_2\}$, $X_3 = \{a_3, c_3\}$, and $X_4 = \{a_4, c_4\}$. Assume the preferences of the DM can be represented by the following utility function:

$$u(x) = u_1(x_1) + u_2(x_1,x_2) + u_3(x_2,x_3) + u_4(x_3,x_4),$$

where the u_i's are given by the tables below:

x_1	a_1	c_1	b_1
$u_1(x_1)$	0	500	1000

$u_2(x_1,x_2)$	a_2	c_2	b_1
a_1	0	10	70
c_1	50	10	90
b_1	60	80	100

$u_3(x_2,x_3)$	a_3	c_3
a_2	0	7
c_2	5	2
b_2	9	10

$u_4(x_3,x_4)$	a_4	c_4
a_3	0	0.6
c_3	0.4	1

Note that the big-stepped structure of utilities in the above tables is consistent with the Ceteris Paribus assumption about preferences, hence $u(\cdot)$ can be characterized by the UCP-net of Figure 8. Asking the DM to provide probability

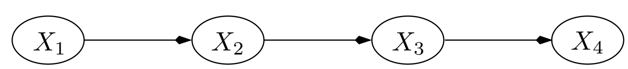


Figure 8: A simple UCP-net.

p such that $c_1 \sim \langle p, b_1; 1-p, a_1\rangle$ would, at first sight, be meaningful and, assuming $u_1(a_1) = 0$ and $u_1(b_1) = 1000$, it would certainly imply that $u_1(c_1) = 1000p$. However, a careful examination highlights that it is not so obvious. Indeed, such gamble, involving only attribute X_1 would be meaningful only if the DM had a preference relation $\succsim_1$ over X_1 that could be exploited to extract informations about $\succsim$, the DM's preference relation over $\mathcal{X}$. In the classical framework of additive conjoint measurement (Fishburn 1970; Krantz *et al.* 1971; Wakker 1989), this property holds because $c_1 \sim \langle p, b_1; 1-p, a_1\rangle$ is equivalent to $(c_1,x_2,x_3,x_4) \sim \langle p, (b_1,x_2,x_3,x_4); 1-p, (a_1,x_2,x_3,x_4)\rangle$ for any $(x_2,x_3,x_4) \in X_2 \times X_3 \times X_4$, but this does not hold for GAI decompositions involving intersecting factors. For instance, using the above tables, it is easily seen that, whatever values for X_3 and X_4:

$$(c_1, a_2, x_3, x_4) \sim \begin{cases} \xrightarrow{0.519} (b_1, a_2, x_3, x_4) \\ \xrightarrow{0.481} (a_1, a_2, x_3, x_4) \end{cases}$$

$$(c_1, c_2, x_3, x_4) \sim \begin{cases} \xrightarrow{0.467} (b_1, c_2, x_3, x_4) \\ \xrightarrow{0.533} (a_1, c_2, x_3, x_4) \end{cases}$$

$$(c_1, b_2, x_3, x_4) \sim \begin{cases} \xrightarrow{0.505} (b_1, b_2, x_3, x_4) \\ \xrightarrow{0.495} (a_1, b_2, x_3, x_4) \end{cases}$$

The explanation of this unfortunate property lies in the misleading interpretation we may have of Ceteris Paribus statements: in the above UCP-net, Ceteris Paribus implies that preferences over X_1 do not depend on the values of the other attributes. The observation of the subutility tables confirm this fact: b_1 is preferred to c_1, that is also preferred to a_1. However, the CP property does not take into account the strength of these preferences while the probabilities involved in the lotteries do: whatever the value of X_2, (b_1, x_2) is always preferred to (c_1, x_2), but the DM prefers more (b_1, c_2) to (c_1, c_2) than (b_1, a_2) to (c_1, a_2) and this results in different values of p in gambles. This explains the discrepancy between $c_1 \sim \langle p, b_1; 1-p, a_1 \rangle$ and the same gamble taking into account the other attributes. This discrepancy is not restricted to UCP-net root nodes, it is easily seen that it also occurs for other nodes such as X_2 or X_3.

To conclude, the GAI networks introduced in this paper allow taking advantage of any GAI decomposition of a multiattribute utility function to construct a compact representation of preferences. The efficiency of the proposed elicitation procedure lies both in the relative simplicity of the questions posed and in the careful exploitation of independences between attributes to reduce the number of questions. This approach of preference elicitation is a good compromise between two conflicting aspects: the need for sufficiently flexible models to capture sophisticated decision behaviors under uncertainty and the practical necessity of keeping the elicitation effort at an admissible level. A similar approach might be worth investigating for the elicitation of multiattribute utility functions under certainty. Resorting to GAI networks in this context might also be efficient to elicit subutility functions under some solvability assumptions on the product set.

References

Bacchus, F., and Grove, A. 1995. Graphical models for preference and utility. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*.

Becker, A., and Geiger, D. 2001. A sufficiently fast algorithm for finding close to optimal clique trees. *Artificial Intelligence* 125(1-2):3–17.

Boutilier, C.; Bacchus, F.; and Brafman, R. I. 2001. UCP-networks; a directed graphical representation of conditional utilities. In *Proceedings of Seventeenth Annual Conference on Uncertainty in Artificial Intelligence*, 56–64.

Boutilier, C.; Brafman, R. I.; Hoos, H. H.; and Poole, D. 1999. Reasoning with conditional ceteris paribus preference statements. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*.

Buntine, W. 1994. Operations for learning with graphical models. *Journal of Artificial Intelligence Research* 2:159–225.

Chajewska, U., and Koller, D. 2000. Utilities as random variables: Density estimation and structure discovery. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*.

Chajewska, U.; Koller, D.; and Parr, R. 2000. Making rational decisions using adaptive utility elicitation. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*.

Cowell, R.; Dawid, A.; Lauritzen, S.; and Spiegelhalter, D. 1999. *Probabilistic Networks and Expert Systems*. Statistics for Engineering and Information Science. Springer-Verlag.

Darwiche, A., and Hopkins, M. 2001. Using recursive decomposition to construct elimination orders, jointrees, and dtrees. In *Proceedings of ECSQARU'01*, 180–191.

Dubois, D., and Prade, H. 1995. Possibility theory as a basis for qualitative decision theory. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 1925–1930.

Fishburn, P. C. 1970. *Utility Theory for Decision Making*. NewYork: Wiley.

Halpern, J., and Chu, F. 2003a. Great expectations. part I: On the customizability of generalized expected utility. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 291–296.

Halpern, J., and Chu, F. 2003b. Great expectations. part II: Generalized expected utility as a universal decision rule. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 297–302.

Heckerman, D. 1995. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research.

Howard, R. A., and Matheson, J. E. 1984. Influence diagrams. In Howard, R. A., and Matheson, J. E., eds., *Readings on the Principles and Applications of Decision Analysis*, volume 2. Strategic Decisions Group. 719–762.

Jensen, F. V.; Lauritzen, S. L.; and Olesen, K. G. 1990. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quarterly* 4:269–282.

Jensen, F. 1996. *An introduction to Bayesian Networks*. Taylor and Francis.

Keeney, R. L., and Raiffa, H. 1976. *Decisions with Multiple Objectives - Preferences and Value Tradeoffs*. Cambridge university Press.

Kjærulff, U. 1990. Triangulation of graphs — algorithms giving small total state space. Technical Report R-90-09, Dept. of Mathematics and Computer Science, Aalborg University.

Krantz, D. H.; Luce, R. D.; Suppes, P.; and Tversky, A. 1971. *Foundations of Measurement (Additive and Polynomial Representations)*, volume 1. New York: Academic Press.

Olesen, K., and Madsen, A. 2002. Maximal prime subgraph decomposition of bayesian networks. *EEE Transactions on Systems, Man and Cybernetics, Part B* 32(1):21–31.

Rose, D. 1970. Triangulated graphs and the elimination process. *Journal of Mathematical Analysis and Applications* 32:597–609.

Savage, L. J. 1954. *The foundations of statistics*. Dover.

Schmeidler, D. 1986. Integral representation without additivity. In *Proceedings of the American Mathematical Society*, volume 97, 255–261.

Shachter, R. 1986. Evaluating influence diagrams. *Operations Research* 34:871–882.

Shafer, G. 1996. *Probabilistic expert systems*. ociety for Industrial and Applied Mathematics.

von Neumann, J., and Morgenstern, O. 1944. *Theory of Games and Economic Behaviour*. Princetown, New Jersey: Princetown University Press.

Wakker, P. P. 1989. *Additive Representations of Preferences, A New Foundation of Decision Analysis*. Dordrecht: Kluwer Academic Publishers.