Easy and optimal queries to reduce set uncertainty

N. Ben Abdallah, S. Destercke, Mohamed Sallak

Easy and optimal queries to reduce set uncertainty: application to reliability

N.Ben Abdallah[a], S.Destercke[b], M.Sallak[b],

[a] *Agroparistech, INRA, MIA518*
[b] *Université de Technologie de Compiègne, CNRS, UMR 7253 Heudiasyc*

Abstract

In this paper, we address the problem of optimally querying experts to reduce set uncertainty. We propose optimal querying strategies for two particular query formats (local bound and pairwise comparisons) based on two selection criteria (the minmax and the Bayesian rules). We study the computational aspects of the optimal solution in the general case and for the specific monotonic and multilinear functions. The use of the proposed approach is illustrated in a common estimation problem in reliability analysis.

Keywords: uncertainty reduction, optimal query, myopic, minimax decision strategy, pairwise comparisons, local bound queries.

1. Introduction

Epistemic uncertainty, which stems from a lack of knowledge, prevails in many areas and applications including risk analyses, machine learning, etc. Contrary to the aleatory uncertainty which is inherent to the system and thereby irreducible, epistemic uncertainty can be reduced through further data. However, in many situations, acquiring experimental data may be costly, time-consuming or technically impossible. In such situations, expert opinions can be a valuable way to enhance the state of knowledge and hence improve the efficiency of a decision or the reliability of an estimation in an uncertain context.

Elicitation - the process of querying experts to acquire information - has been substantially addressed in the probabilistic framework, where an information source - the expert - is asked some generic questions, the answers to which guide the analyst in encoding the expert judgement by probabilities. Elicitation finds also application in the development of decision support systems, or in active learning [24] where the analyst interactively queries a source to label some data to be used in the learning process.

Information acquisition through expert querying often has to be conducted with limited resources (expert availability, number of queries, focusing capability of the expert). Therefore, for the elicitation to be effective, careful consideration should be paid to the choice of the queries to ask. First, both the

questions and its possible answers need to be easily understandable, so that the expert can focus on providing an informed and reliable answer. Second, to get the most of the elicitation, queries should be as informative as possible on the quantity of interest, i.e., they should selected in an optimal way.

How to choose sequences of optimal questions to ask, or even the notion of optimal queries, has received little attention in the literature. Indeed, the great majority of elicitation techniques prescribe generic questions, without considering the consequences of answers on some final goal [2] (the work of Curtis and Wood [9], settled in a probabilistic context, is an exception). Similar issues have however been addressed in other problems such as preference elicitation with pioneering works dating back ten years ago [29, 7] and still thriving today [27, 4, 6]. In the last few years, the use of similar ideas also developed in active learning [1].

The goal of this paper is to explore similar ideas and propose optimal querying strategies that allow to reach a given objective in an effective way. In particular, we are concerned with the objective of reducing epistemic uncertainty expressed by means of sets. The querying strategies we propose are adaptive and optimal, i.e., they select, in real time and based on the previous elicited information, the next best query to ask.

This paper is structured as follows: we first give a general formalization of the problem in Section 2, before exploring two particularly interesting types of queries in Sections 3 and 4. These queries, respectively named local bound and pairwise comparisons, are simple enough to be easily understandable and to offer computational advantages. Section 5 then explores some specific families of functions for which these computational advantages become even stronger. Finally, we apply our results to reliability analysis problems in Section 6.

2. Formal statement of the problem

Let Φ be a continuous function mapping a set of n logically independent real-valued inputs $(x_1, x_2, ..., x_n)$ to a real-valued output $y \in \mathbb{R}$. We assume that, initially, we know the values $(x_1, x_2, ..., x_n)$ to lie in a convex space $\mathcal{X}^0 \subset \mathbb{R}^n$, so that

$$\begin{array}{ccccc} \Phi & : & \mathcal{X}^0 & \to & \mathbb{R} \\ & & \mathbf{x} = (x_1, ..., x_n) & \mapsto & \Phi(\mathbf{x}) = y. \end{array}$$

Denoting by $X_i = \left[\underline{X}_i, \overline{X}_i\right] \subset \mathbb{R}$ intervals for x_i ($i \in N = \{1, ..., n\}$), we will consider the case $\mathcal{X}^0 = \times_{i=1}^n X_i^0$, where X^0 are the initial intervals. Given an interval X, a natural way to quantify the amount of uncertainty in X is by its width

$$U_X = \overline{X} - \underline{X}.$$

Since Φ is continuous, the initial response y corresponding to the initial state of knowledge on the inputs lies in the bounded interval:

$$Y^0 = \Phi(\mathcal{X}^0) = [\min_{\mathbf{x} \in \mathcal{X}^0} \Phi(\mathbf{x}), \max_{\mathbf{x} \in \mathcal{X}^0} \Phi(\mathbf{x})] = [\underline{Y}^0, \overline{Y}^0]. \quad (1)$$

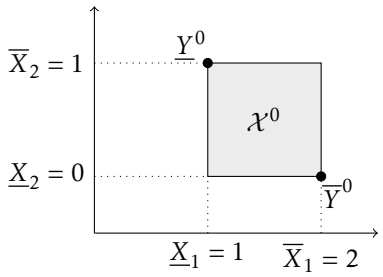


Figure 1: Example 1: initial space $\mathcal{X}^0$ and bounds of Φ.

Depending on Φ, computing the (exact) bounds of Equation (1) can already be a quite challenging problem. Section 5 discusses this issue in more details, and investigates specific cases of practical interest for which this problem is tractable.

Example 1. *We consider a very simple function $y = \phi(x_1, x_2) = x_1 - x_2$, and where $\mathcal{X}^0 = [1,2] \times [0,1]$ ($X_1^0 = [1,2]$ and $X_2^0 = [0,1]$). From this, we have*

$$Y^0 = [\Phi(\underline{X_1}, \overline{X_2}), \Phi(\overline{X_1}, \underline{X_2})] = [\underline{Y}^0, \overline{Y}^0] = [0,2]$$

from which $U_{Y^0} = \overline{Y}^0 - \underline{Y}^0 = 2$. The situation is depicted by Figure 1.

The problem we are considering is the following: we want to reduce our uncertainty $U_Y = \overline{Y} - \underline{Y}$ by asking question to experts about the input variables x_i, to attain some objectives. For instance, we may want to reduce the uncertainty under some threshold $U_Y \leq s_0$ or simply reduce the most U_Y in a given number of questions. To achieve this, we will iteratively refine the input space $\mathcal{X}^k$ into $\mathcal{X}^{k+1} \subset \mathcal{X}^k$ (starting from $\mathcal{X}^0$) by asking questions to an expert.

As expert elicitation is time-consuming and cognitively demanding for the expert, and economically expensive for the decision maker, we want to ask as few questions as possible, or to be the most effective possible on those questions we ask. In other words, we want the querying strategy to be optimal. This is what we develop in the next sections.

2.1. Queries and answers

Formally, a query Q will come from a space $\mathcal{Q}$ of possible queries. Each query can then receive an answer A lying in a space $\mathcal{A}$[1].

At a given iteration $(k+1)$ of the elicitation process, a query and its answer (Q, A) will define a subspace of $\mathcal{X}^k$, and we will also use (Q, A) to denote this

[1] To simplify notations, the set of answers is assumed to be the same for all queries. Extension of the framework to query-dependent answers can easily be done.

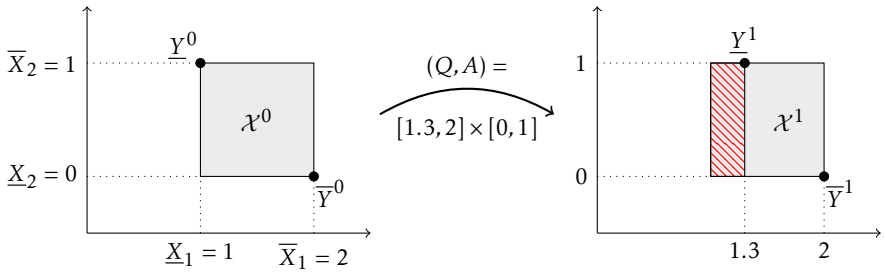


Figure 2: Updated space $\mathcal{X}^1$ of Example 2.

subspace. The result of an iteration will be the reduced space

$$\mathcal{X}^{k+1} = \mathcal{X}^k \cap (Q, A) := \mathcal{X}^k(Q, A).$$

Once $\mathcal{X}^k$ is updated into $\mathcal{X}^{k+1}$, we can compute the resulting interval $Y^{k+1} := Y^k(Q, A)$ using Equations (1) with the index 0 replaced by $k+1$. Note that, since $\mathcal{X}^{k+1} \subset \mathcal{X}^k$, we do have that $Y^{k+1} \subseteq Y^k$ and

$$U_{Y^k} \geq U_{Y^k(Q,A)}, \tag{2}$$

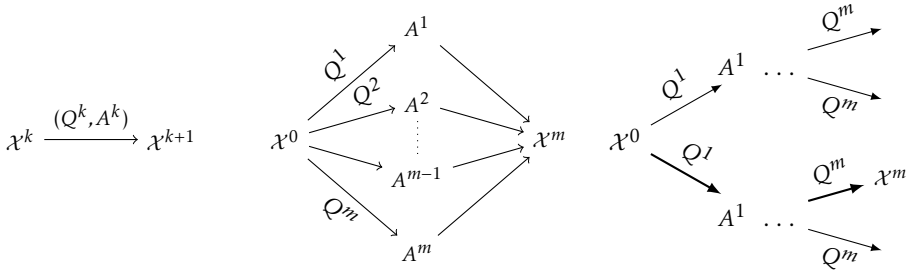
which ensures that uncertainty will decrease after each question.

In general expert elicitation, it is important to use simple questions so that the expert can focus her/his cognitive effort on answering the question rather than understanding it. This is particularly true when the elicitation is made of many successive questions. In the rest of the paper, we will consider two simple question types: local bound queries ("$x_i \leq \alpha$?", $i \in N$, $\alpha \in X_i$) and pairwise comparison judgements ("$x_i \leq x_j$"?, $(i, j) \in N \times N \setminus \{i\}$). Both types are quite natural in many applications, such as reliability or risk analysis where it is common to ask experts to rank some variables of interest or compare them to reference thresholds. In addition to being easily understandable, these queries are easy to respond as the possible answer is either Yes or No. Given a question of the type $Q = \{x_i \leq \alpha\}$? and the answer $A = Yes$, then the associated subspace is $(Q, A) = \{\times_{j=1}^{i-1} X_j^0 \times [\underline{X}_i^0, \alpha] \times_{j=i+1}^{n} X_j^0\}$.

Example 2. *Let us pursue Example 1 and consider the question* $Q = \{x_1 \leq 1.3\}$ *with a negative answer, we then have* $\mathcal{X}^1 = [1.3, 2] \times [0, 1]$ *and* $Y^1 = [0.3, 2]$

Note that in this paper we assume the expert to be an oracle, so the "I don't know" answer is not considered here[2].

[2]Should the expert return "I don't know" to Q, then a simple heuristic strategy is to remove Q from the question set $\mathcal{Q}$ and then select the optimal one among the remaining ones.



(a) Myopic (b) Simultaneous (c) Sequential

Figure 3: Schemas of three approaches to optimal querying.

2.2. Query selection strategies

When the analyst has a set of candidate questions $\mathcal{Q}$ with possible answers in $\mathcal{A}$, a query selection strategy consists in defining the best question to ask at each iteration of the elicitation process. There are at least three main ways to do so:

- **myopically**, where at each iteration a single next optimal question is selected and presented to the expert;
- **simultaneously**, where the analyst knows that he has a "budget" of m questions, and search for the m optimal questions to ask, without assuming that the questions will be asked sequentially (they may even be answered by different experts);
- **sequential**, where the analyst knows that he has a "budget" of m questions, and can determine in which sequence they will be presented. In contrast with the simultaneous approach, the jth question can depend on the observed answers so far.

These three frameworks are ordered in terms of increasing complexity, and can be somehow mixed: for instance, one could consider a "sequential-myopic" approach where each iteration consists of two sequential questions. Those different approaches are sketched in Figure 3.

This work studies the myopic approach, for the following reasons: it is often simpler to solve, sometimes allowing for analytical exact solutions, and does not require to specify the number of asked questions in advance, a particularly interesting feature in iterative and interactive querying process.

As the goal of the elicitation is to reduce the uncertainty on the system output Y, the myopic approach, at step k, should search for the question Q maximizing the uncertainty reduction in the output Y^{k+1}, or equivalently minimizing the resulting $U_{Y^{k+1}}$. However, the interval Y^{k+1} resulting from Q is uncertain, as we do not know what the answer to Q will be. We can express the optimization problem as follows:

$$Q^* = \arg\min_{Q \in \mathcal{Q}} U^{cr}_{Y^k(Q,\cdot)}, \tag{3}$$

where cr stands for the criterion we use to evaluate the value of question Q, and $U^{cr}_{Y^k(Q,\cdot)}$ is the potential uncertainty reduction induced by query $Q \in \mathcal{Q}$. Decision theory proposes several rules to make choices in an uncertain context including:

- Minimin (mm), which reads

$$U^{mm}_{Y^k(Q,\cdot)} = \min_{A \in \mathcal{A}} U_{Y^k(Q,A)}. \tag{4}$$

  It corresponds to an extreme optimistic attitude, as it selects the alternative with the best outcome (the answer A minimizing $U_{Y^k(Q,A)}$).

- Minimax (mM) criterion [28]

$$U^{mM}_{Y^k(Q,\cdot)} = \max_{A \in \mathcal{A}} U_{Y^k(Q,A)}. \tag{5}$$

  It is a pessimistic criterion that selects the alternative with the worst outcome (the answer A maximizing $U_{Y^k(Q,A)}$). While it may be too conservative in some situations, it guarantees a minimal improvement.

- Hurwicz's criterion ($H(q)$), that reads

$$U^{H(q)}_{Y^k(Q,\cdot)} = (1-q) \max_{A \in \mathcal{A}} U_{Y^k(Q,A)} + q \min_{A \in \mathcal{A}} U_{Y^k(Q,A)}. \tag{6}$$

  It is a trade-off between the optimistic and pessimistic strategies, where q is a coefficient reflecting the decision maker's degree of optimism.

- Bayesian (B) approach: it assumes the existence of a probability distribution P over the realizations of the uncertain event (here the answer) and evaluates a decision by its expected outcome:

$$U^{B}_{Y^k(Q,\cdot)} = \mathbb{E}_P(U_{Y^k(Q,A)}). \tag{7}$$

In what follows, we will focus mainly on the minimax and Bayesian selection strategies, that are by far the most popular in solving optimization problems, and present computational as well as theoretical advantages for large families of functions (this was already shown in a previous paper [3] in which sketch of proofs were presented for the local bound queries). For both query types, we begin by evaluating the contribution of a couple $(Q,A) \in \mathcal{Q} \times \mathcal{A}$ to the total uncertainty reduction on Y, then we characterize the optimal query according to the selected criteria.

3. Local bound queries

Local bound queries ask to compare the value of the variable x_i to some reference value α. We denote the set of local bound queries of type "$x_i \leq \alpha$?", with $\alpha \in X_i$ by $\mathcal{Q}^{LB} = \left\{Q_i^\alpha, i \in \mathcal{N}, \alpha \in X_i\right\}$ and the set of answers $\mathcal{A} = \{Yes, No\}$.

3.1. The contribution of a bound query

When a question Q_i^α is asked and an answer $A \in \mathcal{A}$ is given, then the corresponding subspace is

$$(Q_i^\alpha, A) = \begin{cases} \{\times_{j=1}^{i-1} X_j^0 \times [\underline{X}_i^0, \alpha] \times_{j=i+1}^{n} X_j^0\} & \text{if } A = Yes \\ \{\times_{j=1}^{i-1} X_j^0 \times [\alpha, \overline{X}_i^0] \times_{j=i+1}^{n} X_j^0\} & \text{if } A = No. \end{cases} \tag{8}$$

If only local bound queries are used, then the starting hypercube $\mathcal{X}^0 = \times_{i=1}^n X_i^0$ will remain a hypercube after each query. In this case, the space $\mathcal{X}^{k+1}$ resulting from the combination of $\mathcal{X}^k$ and the answer to Q_i^α is $\times_{i=1}^n X_i^{k+1}$ with

$$X_j^{k+1} = \begin{cases} X_j^k & \text{if } i \neq j \\ X_j^k \cap [-\infty, \alpha] & \text{if } i = j, A = Yes \\ X_j^k \cap [\alpha, +\infty] & \text{if } i = j, A = No. \end{cases} \tag{9}$$

Example 3. *Consider the function* $\Phi(x_1, x_2, x_3) = x_1 x_2 - x_2 x_3$ *with* $X_1^0 = X_3^0 = [0, 1]$, $X_2^0 = [0.2, 1]$, *then we have:*

$$\underline{Y}^0 = \Phi(\underline{X}_1^0, \overline{X}_2^0, \overline{X}_3^0) = -1; \overline{Y}^0 = \Phi(\overline{X}_1^0, \overline{X}_2^0, \underline{X}_3^0) = 1$$

i.e.,

$$U_{Y^0} = 2.$$

Assume now we ask the question $Q_1^{0.5}$ *and receive the answer Yes, then*

$$X_1^1 = [0, 0.5],$$

$$\overline{Y}^1 = \Phi(\overline{X}_1^1, \overline{X}_2^1, \underline{X}_3^1) = 0.5,$$

the lower bound $\underline{Y}^1 = \underline{Y}^0$ *and*

$$U_{Y^1} = 1.5.$$

3.2. Optimal query selection

When local bound queries are considered, the $(k+1)$th optimal query to be selected in Equation (3) can be written as:

$$Q^* = (i^*, \alpha^*) = \arg\min_{i \in N} \min_{\alpha \in X_i^k} U^{cr}_{Y^k(Q_i^\alpha, \cdot)}, \tag{10}$$

where X_i^k denotes the projection of $\mathcal{X}^k$ on the ith dimension. Equation (10) is a two stage optimization problem: we first determine the optimal local bound value for each input i, and calculate the uncertainty reduction induced by that

Algorithm 1: Iterative elicitation for uncertainty reduction (bound queries).

$Inputs : \mathcal{X}^0, s_0, k = 0$
while $U_{Y^k} >= s_0$ **do**
  for i in N **do**
    Compute $\alpha^* = \arg\min_{\alpha \in X_i^k} U^{cr}_{Y^k(Q_i^\alpha, \cdot)}$
    Compute $U^{cr}_{Y^k(Q_i^{\alpha^*}, \cdot)}$
  end for
  $i^* = \arg\min_{i \in N} U^{cr}_{Y^k(Q_i^{\alpha^*}, \cdot)}$
  Ask query: "$x_{i^*} <= \alpha^*$?"
  Obtain answer A
  Update $\mathcal{X}^k$ into $\mathcal{X}^{k+1} = \mathcal{X}^k(Q_{i^*}^{\alpha^*}, A)$
  Compute $U_{Y^{k+1}}$
  k=k+1
end while

local query; we then select the element i^* that leads to the highest uncertainty reduction in y (Algorithm 1).

The major difficulty of Algorithm 1 is the evaluation of α^* for each possible variable x_i. In what follows, we study in detail the computational complexity of computing the optimal bound α^* for the minimax and Laplacian criteria, and discuss briefly the minimin strategy.

3.2.1. Minimax strategy

For the minimax strategy, finding α^* for a given x_i at a step $(k+1)$ reads

$$\alpha^{*,mM} = \arg\min_{\alpha \in X_i^k} \max(U_{Y^k(Q_i^\alpha, No)}, U_{Y^k(Q_i^\alpha, Yes)}) \tag{11}$$

The next proposition shows that solving the optimization problem (11) to get the $\alpha^{*,mM}$ is equivalent to finding the intersection of the two functions $U_{Y(Q_i^\alpha, Yes)}$, $U_{Y(Q_i^\alpha, No)}$ of α. In practice, a general and efficient method to find the solution is then to use a dichotomy search on the space $[\underline{X}_i, \overline{X}_i]$.

Proposition 1. *Functions $U_{Y^k(Q_i^\alpha, Yes)}$ and $U_{Y^k(Q_i^\alpha, No)}$ measuring the uncertainty level on Y^k induced by a positive and a negative answer to Q_i^α and defined on X_i^k*

- *are increasing and decreasing in α, respectively, and*
- *always have a non-empty intersection $M_i \subset X_i^k$ such that*

$$U_{Y^k(Q_i^\alpha, Yes)} = U_{Y^k(Q_i^\alpha, No)}$$

for any $\alpha \in M_i$ (M_i is a single point or an interval).

Proof. *We have that*

$$U_{Y^k(Q_i^\alpha, Yes)} = \max_{\mathcal{X}^k(Q_i^\alpha, Yes)} \Phi(\mathbf{x}) - \min_{\mathcal{X}^k(Q_i^\alpha, Yes)} \Phi(\mathbf{x}) \tag{12}$$

and

$$U_{Y^k(Q_i^\alpha, No)} = \max_{\mathcal{X}^k(Q_i^\alpha, No)} \Phi(\mathbf{x}) - \min_{\mathcal{X}^k(Q_i^\alpha, No)} \Phi(\mathbf{x}) \tag{13}$$

To show that $U_{Y^k(Q_i^\alpha, Yes)}$ is increasing in α, we need to show that $U_{Y^k(Q_i^\alpha, Yes)} \leq U_{Y^k(Q_i^\beta, Yes)}$ for $\alpha \leq \beta$. This result follows from the fact that $(Q_i^\alpha, Yes) \subseteq (Q_i^\beta, Yes)$, since

$$\{\times_{j=1}^{i-1} X_j^0 \times [\underline{X}_i^0, \alpha] \times_{j=i+1}^{n} X_j^0\} \subseteq \{\times_{j=1}^{i-1} X_j^0 \times [\underline{X}_i^0, \beta] \times_{j=i+1}^{n} X_j^0\}.$$

The same reasoning can be applied to $U_{Y^k(Q_i^\alpha, No)}$ to show that it is decreasing in α.

To demonstrate the second part of the proposition, simply observe that :

$$\max_\alpha U_{Y^k(Q_i^\alpha, Yes)} = U_{Y^k(Q_i^{\overline{X}_i^k}, Yes)} = U_{Y^k},$$

$$\max_\alpha U_{Y^k(Q_i^\alpha, No)} = U_{Y^k(Q_i^{\underline{X}_i^k}, No)} = U_{Y^k},$$

where U_{Y^k} is the uncertainty before the question. These two results stem from the fact that answering Yes to the question $x_i <= \overline{X}_i^k$ simply brings no information (and similarly for No). As both functions have the same maximum, are continuous (since Φ is), and are respectively increasing and decreasing in α, they have at least one point of intersection.

When M_i is an interval $[\underline{M}_i, \overline{M}_i]$, we simply take the middle point $\alpha^* = \underline{M}_i + \overline{M}_i/2$. This never occurs when one of the two functions $U_{Y^k(Q_i^\alpha, Yes)}$, $U_{Y^k(Q_i^\alpha, No)}$ is strictly monotonic in α.

In some situations, it may also happen that the intersection occurs on one of the bounds of X_i^k for all i, which means that the proposed optimal question will probably be uninformative, unless the expert answer reduces the interval $[\underline{X}_i, \overline{X}_i]$ to a point, an unlikely scenario. When this occurs, we use a different strategy that defines optimality as the highest reduction of uncertainty, no more on y, but on x_i. This heuristic is equivalent, when X_is are intervals, to choosing the largest interval $i^* = \arg\max_i U_{X_i^k}$ and to pick the mid of this interval, i.e., $\alpha^* = (\underline{X}_{i^*}^k + \overline{X}_{i^*}^k)/2$.

Remark 1. *Since $U_{Y^k(Q_i^\alpha, Yes)}$ and $U_{Y^k(Q_i^\alpha, No)}$ are respectively increasing and decreasing functions of α, this means that for a given variable x_i, the value $\alpha^{*,mm} = \arg\min_{\alpha \in X_i^k} \min(U_{Y^k(Q_i^\alpha, No)}, U_{Y^k(Q_i^\alpha, Yes)})$ will be obtained either for $\alpha^{*,mm} = \underline{X}_i^k$ or $\alpha^{*,mm} = \overline{X}_i^k$. These corresponds to very extreme questions, reducing the interval X_i^k either to a point (a quite unlikely outcome in practice) or not changing it at all. This means that the minimin approach offers little interest in the case of local bound queries.*

3.2.2. Bayesian strategy

In the minimax approach, we do not use any a priori information about the likelihood of answering Yes or No. However, this can lead to consider very unlikely answers, such as answering Yes to $Q_i^{\underline{X}_i}$ (as is the case in the minimin strategy, see Remark 1). Another solution is then to consider a probabilistic strategy, where we assume the existence of a probability distribution P over the set of answers $\mathcal{A}$, this probability modelling our subjective beliefs about the likelihood of getting the different answers. We then evaluate a query Q by the expected reduction $\mathbb{E}_P(U_{Y^k(Q,\cdot)})$ of uncertainty on y induced by the possible answers:

$$U^B_{Y^k(Q_i^\alpha,\cdot)} = \mathbb{E}_P(U_{Y^k(Q_i^\alpha,\cdot)}) = \sum_{A\in\mathcal{A}} P(A) U_{Y(Q_i^\alpha,A)} \tag{14}$$

$$\alpha^{*,B} = \arg\min_{\alpha\in X_i^k} (P(Yes|Q_i^\alpha) U_{Y^k(Q_i^\alpha,Yes)} + P(No|Q_i^\alpha) U_{Y^k(Q_i^\alpha,No)}). \tag{15}$$

When the available evidence suggests that a quantity x_i lies in an interval X_i, it is common to follow Laplace's indifference principle and quantify uncertainty by assuming a uniform probability distribution over that set. Under this assumption, the probability of the positive and negative answers to a question Q_i^α are proportional to the width of the sub-interval of X_i they lead to:

$$P(Yes|Q_i^\alpha) = P(\underline{X}_i \le x_i \le \alpha) = \frac{\alpha - \underline{X}_i}{\overline{X}_i - \underline{X}_i}$$

and

$$P(No|Q_i^\alpha) = P(\alpha \le x_i \le \overline{X}_i) = \frac{\overline{X}_i - \alpha}{\overline{X}_i - \underline{X}_i}.$$

These probabilities can then be modified according to the information we have (for instance, if we have reasons to think that the true value is closer to one of the bounds $\underline{X}_i$ or $\overline{X}_i$).

Remark 2. *Note that when $U_{Y^k(Q_i^\alpha,Yes)} = U_{Y^k(Q_i^\alpha,No)} = U_{Y^k(Q_i^\alpha)}$, then $\mathbb{E}_P(U_{Y^k(Q_i^\alpha,A)}) = U_{Y^k(Q_i^\alpha)}$, whatever the values of P. This means that the function $U^B_{Y^k(Q_i^\alpha)}$ has value $U^B_{Y^k(Q_i^{\alpha^{*,mM}})} = U^{mM}_{Y^k(Q_i^{\alpha^{*,mM}})}$, since the minimax is obtained at the intersection M_i of $U_{Y^k(Q_i^\alpha,Yes)}$ and $U_{Y^k(Q_i^\alpha,No)}$. This means that we have*

$$\min_{X_i^k} U^B_{Y^k(Q_i^\alpha,\cdot)} = U^B_{Y^k(Q_i^{\alpha^{*,B}})} \le U^{mM}_{Y^k(Q_i^{\alpha^{*,mM}})} = \min_{X_i^k} U^{mM}_{Y(Q_i^\alpha,\cdot)},$$

hence the expected uncertainty reduction with a Bayesian strategy is at least as high as the one obtained by the minimax strategy. However, in contrast with this latter, the Bayesian strategy does not offer guarantees about the uncertainty reduction, in the sense that the actual reduction may be lower than the expected one.

Also, while Proposition 1 indicates that $\alpha^{*,mM}$ can be obtained by a dichotomic search, this cannot be done in general for the Bayesian strategy, which therefore potentially requires heavier computations.

Example 4. *Consider the function* $\Phi(x) = x_1x_2 - x_2x_3 + x_2$ *where* $X_1^0 = X_3^0 = [0, 0.5]$ *and* $X_2^0 = [0, 1]$. *Figure 4 shows the various strategies for* Q_2^α. *We can see that the minimax and the Laplacian strategies recommend respectively* $\alpha^{*,mM} = 3/4$ *and* $\alpha^{*,B} = 1/2$. *Another remark is that* $U_{Y^0(Q_2^\alpha,Yes)}$ *and* $U_{Y^0(Q_2^\alpha,No)}$ *are both linear. We will see in the next section that this is true for monotonic multi linear functions.*

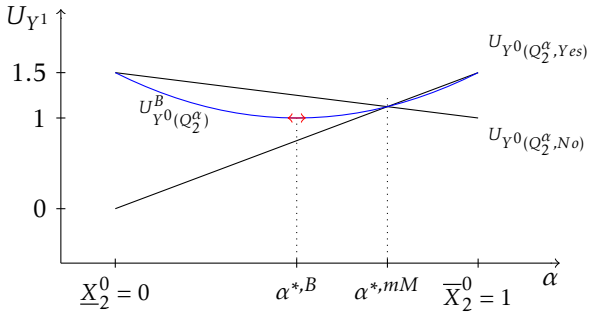


Figure 4: Optimal recommendations of the minimax and Bayesian strategies.

We have seen that elicitation based on local bound queries leads to computationally tractable computations, provided we can evaluate bounds $\underline{Y}^k, \overline{Y}^k$ efficiently. However, as it involves only one variable per iteration, it may lead to many questions in high dimensional problems. It also requires the expert to compare x_i with a numerical bound, which may be difficult in some cases. These two reasons led us to investigate pairwise comparisons queries, that update the state of knowledge on more than one variable per iteration, and do not involve numerical bounds explicitly. As they remain simple, they are an appealing alternative or complement to local bound queries.

4. Pairwise comparison judgements

Pairwise comparisons simply ask to compare the values of two variables, that is to ask for a pair (i, j) whether "$x_i \leq x_j$?". We denote such a query by $Q_{i,j}$. As we assume that each variable has an ill-known yet precise value and that the n variables describing the system are comparable. The set of pairwise queries is finite and contains at most $n(n-1)/2$ elements corresponding to the pairs $\{(i, j) \in \{1, \ldots, n\} \times \{i+1, \ldots, n\}\}$.

The set of relevant queries is the subset of pairs (i, j) such that no evidence exists yet on their ordering. Finding this set can easily be done by encoding the pairwise comparisons already known at step k of the elicitation process in an adjacency matrix - a $n \times n$ Boolean matrix M^k defined by:

$$M^k(i,j) = \begin{cases} 1 & \text{if } x_i \leq x_j, j \neq i \\ 0 & \text{elseif.} \end{cases} \tag{16}$$

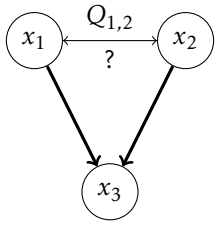


Figure 5: Example of partial ordering between variables.

The set of relevant queries is then given by

$$\mathcal{Q}^C = \left\{(i,j) \in \{1,\dots,n\} \times \{i+1,\dots,n\} : M^{k,+}(i,j) = M^{k,+}(j,i) = 0\right\}$$

where $M^{k,+}$ is the transitive closure [3] of M^k. $M^{k,+}$ can easily be computed using the Floyd-Warshall algorithm [30], which has a $\mathcal{O}(n^3)$ complexity.

The initial matrix M^0 is such that $M^0(i,j) = 1$ if $x_i \leq x_j$ for all $\mathbf{x} \in \mathcal{X}^0$. If furthermore $\mathcal{X}^0 = \times_{i=1}^n X_i^0$ si a hypercube, then $M^0(i,j) = 1$ if $\overline{X}_i^0 \leq \underline{X}_j^0$, which is very easy to check.

Example 5. *Consider the case where* $X_1^0 = [1.3, 3]$, $X_2^0 = [2, 3.5]$, $X_3^0 = [0.6, 1]$. *The matrix* M^0 *is given by*

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

and the only possible comparative query is $\mathcal{Q}^C = \{(1,2)\}$. *The induced partial order on elements is pictured in Figure 5.*

4.1. The contribution of a comparative query

Given a query $Q_{i,j} \in \mathcal{Q}^C$ and an answer A, then

$$(Q_{i,j}, A) = \begin{cases} \left\{\mathbf{x} \in \mathcal{X}^0 | x_i \leq x_j\right\} & \text{if } A = Yes \\ \left\{\mathbf{x} \in \mathcal{X}^0 | x_i \geq x_j\right\} & \text{if } A = No, \end{cases} \tag{17}$$

and $\mathcal{X}^k \cap (Q_{i,j}, A)$ is no longer a hypercube, as $(Q_{i,j}, A)$ consists of cutting space $\mathcal{X}^k$ into two half spaces along equation $x_i = x_j$, and then to consider one of these half spaces (corresponding to the inequality induced by A). The implication of $\mathcal{X}^k$ being no longer a hypercube but a convex polyhedron will be discussed in Section 5.

[3] The transitive closure of a binary relation R defined on a set $\mathcal{X}$ is the transitive relation R' on $\mathcal{X}$ such that R' encloses R and is minimal.

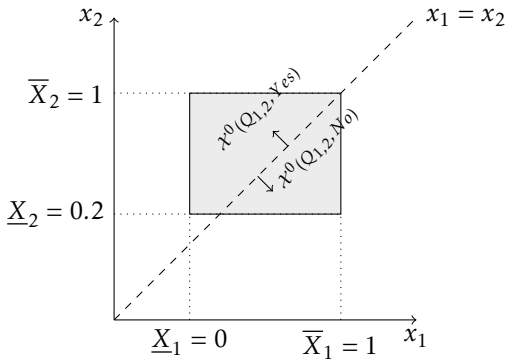


Figure 6: Projection of $X(Q_1^2, Yes)$ and $X(Q_1^2, No)$;

At a given step k+1 where $Q_{i,j}$ is asked, the adjacency matrix is also updated into

$$\begin{cases} M^{k+1}(\ell, f) = 1 & \text{if } \ell = i, f = j, A = Yes; \quad (18a) \\ M^{k+1}(\ell, f) = 1 & \text{if } \ell = j, f = i, A = No; \quad (18b) \\ M^{k+1}(\ell, f) = M^k(\ell, f) & \text{else.} \quad (18c) \end{cases}$$

whose transitive closure $M^{k+1,+}$ can again be obtained through Floyd-Warshall algorithm.

Example 6. *Going back to Example 3, Figure 6 depicts geometrically the constraint induced by $Q_{1,2}$.*

If the answer to the query is negative then:

$$\underline{Y}^1 = \Phi(0.5, 0.5, 1) = -0.25; \overline{Y}^1 = \Phi(1, 1, 0) = 1$$

i.e.,

$$U_{Y^1} = 1.25.$$

4.2. Query selection strategy

The set $\mathcal{Q}^C$ of comparative queries being finite, finding the optimal pair in Equation (3) can be achieved via a brute force search that first calculates the uncertainty reduction on Y induced by every $(i, j) \in \mathcal{Q}^C$, and then picks the one that leads to the highest value. Algorithm 2 describes the procedure. When the selection strategy leads to two or several equally optimal queries, ties can be broken random.

4.2.1. Minimax strategy

The minimax strategy (4) for comparative queries is given by

$$U^{mM}_{Y^k(Q_{i,j},\cdot)} = \max(U_{Y^k(Q_{i,j},Yes)}, U_{Y^k(Q_{i,j},No)}). \quad (19)$$

Algorithm: Iterative elicitation for uncertainty reduction (comparative queries).

$Inputs : \mathcal{X}^0, s_0, k = 0, M^{0,+}$
while $(U_{Y^k} >= s_0)$ and $(\mathcal{Q}^C \neq \emptyset)$ **do**
  for $(i,j) \in \mathcal{Q}^C$ **do**
    Compute $U^{cr}_{Y^k(Q_{i,j},\cdot)}$
  end for
  $(i^*, j^*) = \text{argmin}_{(i,j)\in\mathcal{Q}^C} U^{cr}_{Y^k(Q_{i,j},\cdot)}$
  Ask query: "$x_{i^*} < x_{j^*}$?"
  Obtain answer A
  Compute $U_{Y^{k+1}} = U_{Y^k(Q^{j^*}_{i^*},A)}$
  Build $M^{k+1,+}$
  Update $\mathcal{Q}^C$
  k=k+1
end while

When the Equation

$$\arg \min_{(i,j)\in\mathcal{Q}^C} U^{mM}_{Y^k(Q_{i,j},\cdot)} \tag{20}$$

has multiple answers, we use a lexicographic ordering between these answers. That is, if $Q_{i,j}$ and $Q_{i',j'}$ are both solutions to Equation (20) (they have the same worst case), then $Q_{i,j}$ is preferred to $Q_{i',j'}$ if

$$\min(U_{Y^k(Q_{i,j},No)}, U_{Y^k(Q_{i,j},Yes)}) < \min(U_{Y^k(Q_{i',j'},No)}, U_{Y^k(Q_{i',j'},Yes)}).$$

4.2.2. Bayesian selection

The minimax strategy (7) for comparative queries is given by

$$U^B_{Y^k(Q_{i,j},\cdot)} = P(Yes|Q_{i,j})U_{Y^k(Q_{i,j},Yes)} + P(No|Q_{i,j})U_{Y^k(Q_{i,j},No)}, \tag{21}$$

where $P(Yes|Q_{i,j})$ (resp. $P(No|Q_{i,j})$), is the ratio of the volume of the polytope $\mathcal{X}^k(Q_{i,j}, Yes)$ (resp. $\mathcal{X}^k(Q_{i,j}, No)$ to the volume of $\mathcal{X}^k$. In Example 6, we have for $P(Yes|Q_{i,j}) > P(No|Q_{i,j})$.

Computing integrals over polytopes is known to be a hard problem. In particular, computing the volume of a polytope defined by its vertices or facets is # P-hard [12, 8], so exact values may not be easy to obtain, especially for high dimensional problems. Computations can however be extremely simplified when the constraints are independent. In the general case, the volumes can be efficiently approached using Monte Carlo integration.

Remark 3. *Pairwise comparisons are quite interesting, as it is sometimes easier to compare two quantities rather than comparing a quantity to a numerical value. It is also easier to search through them, as they are in finite number. However, one of*

their drawback is that they will usually not reduce the set $\mathcal{X}^0$ to a single value, even if the $n(n-1)/2$ comparisons are performed.

Indeed, if the intersection $\cap_{i\in\{1,\dots,n\}} X_i^0 = X_\cap^0 \neq \emptyset$ of all the initial intervals is non-empty, then any set $\mathcal{X}^k$ obtained through pairwise comparisons will always contain the subset $\{\mathbf{x} : x_1 = x_2 = \dots = x_n, x_i \in= X_\cap^0\}$. This also indicates that pairwise comparisons alone will have a limited value if all initial intervals taken together have a large overlap (but may remain quite useful if only some of them have a significant overlap).

The above remark also tells us that it may be interesting to combine both queries.

4.3. Combining both query types

Considering both classes, the query set is given by: $\mathcal{Q} = \mathcal{Q}^C \cup \mathcal{Q}^{LB}$, which still has the same set of answers $\mathcal{A}$. The elicitation procedure is described in the following algorithm.

Algorithm: Iterative elicitation for uncertainty reduction (comparative and LB queries).

$Inputs : \mathcal{X}^0$, s_0, $k = 0$, $M^{0,+}$
while $(U_Y^{cr} >= s_0)$ **do**
  Compute $Q_{k^*}^{\alpha^*}$
  if $\mathcal{Q}^C \neq \emptyset$ **then**
    Compute Q_{i^*,j^*}
  end if
  $Q^* = \text{argmin}_{Q\in\left\{Q_{i^*,j^*}, Q_{i^*}^{\alpha^*}\right\}} (U_{Y^k(Q,\cdot)}^{cr})$
  Ask query: "Q^*?"
  Obtain answer A
  Compute $U_{Y^{k+1}} = U_{Y^k(Q^*,A)}$
  Update $\mathcal{X}^k$, $\mathcal{Q}^C$ and $M^{k,+}$ if necessary.
  k=k+1
end while

From a computational point of view, mixing both query types complicates the computations as soon as a comparative query is performed, resulting in a convex set $\mathcal{X}^k$ that is not a hypercube. In particular, it may complicates the computations of bounds $\underline{Y}^k, \overline{Y}^k$ significantly. This is discussed in the next section for specific functions of interest.

5. Computational aspects for specific cases

In this section, we analyse what become of the computations involved in determining the optimal query at each iteration for specific functions. In particular, we are concerned with multilinear and monotonic functions.

5.1. Multilinear functions

A Multi-linear function over variables $x_1,\ldots,x_n$ is a multivariate polynomial in which the variables are raised to the powers of 0 or 1. It can be written in the following form:

$$\Phi(x_1,...,x_n) = \sum_{A\subseteq N} d_A \prod_{i\in A} x_i \tag{22}$$

with $d_A \in \mathbb{R}$ real-valued coefficients. These functions form a large class that encompasses the linear, quadratic, and bilinear functions. As any pseudo-Boolean function can be rewritten in this form [16], multilinear functions are encountered in problems where pseudo-Boolean functions have a role, such as cooperative game theory [22] (where Boolean functions model the payoff of a coalition of players), multicriteria decision-making [14] (interactions among criteria are modeled by Boolean functions), combinatorial optimization [31], reliability theory [20], etc.

The multilinearity of Φ provides nice computational facilities when the bounds are to be found on a hypercube $\mathcal{X}$. First, it is known [18] that the upper and lower bounds of a non constant multilinear function are reached at the vertices of the n-dimensional hypercube of the variables. Actually, multilinear functions belong to a family of functions that are locally monotone[4] in every variable, and it is known that the bounds of a function are obtained for boundary values of variables in which they are locally monotone. This property makes the bounds computations in Equation 1 tractable, at least for low dimensional problems, since the exact solution can be found (greedily) by enumerating all the extreme points of the joint space ($\mathcal{O}(n2^{2n})$ complexity).

Second, provided that $0 \notin X_i$, the local bound minimax strategy (11) will produce a unique optimal bound $M^* = \alpha^{*,mM}$ due to the fact that $U_{Y(Q_i^\alpha,Yes)}$, $U_{Y(Q_i^\alpha,No)}$ are strictly increasing and decreasing, respectively (as the bounds in Equation (22) will be strictly monotone functions).

When it comes to comparative queries, computing $U_{Y(Q_{i,j},Yes)}$, $U_{Y(Q_{i,j},No)}$ consists in finding the bounds over a multilinear functions in a convex domain. However, as Equation (22) is non-convex, this leads to non-convex optimisation problems, for which no efficient, exact computation techniques exist. Customarily, finding global solutions can be done by considering a convex relaxation of the original problem [21, 23, 25], which computational costs grow exponentially as the number of variables increase.

Example 7. *Consider the function* $\Phi(x_1,x_2,x_3) = x_1x_2 - x_2x_3$ *with* $X_1^0 = X_3^0 = [0,1], X_2^0 = [0.2,1]$.

[4] ϕ is locally monotone in x_i if, all other variables being fixed, it is either decreasing or increasing in x_i. Function Φ of Example 3 is locally monotone in x_2, as it is either increasing or decreasing in x_2 once x_1 and x_3 are fixed (see **??** for details).

Figure 7 displays the solutions Φ over $\mathcal{X}^1 = \mathcal{X}^0(Q_{1,2}, Yes)$ on the plan $x_3 = 1$, and for which we get

$$\underline{Y}^1 = \Phi(0,1,1) = -1; \underline{Y}^1 = \Phi(0.5, 0.5, 1) = -0.25$$

Note that the optimal solution lies on the boundary of the polytope. Actually, this result holds for every multilinear optimisation problem with non negative variables [10, 11]. Hence, if solution lies on a boundary for which $x_i = x_j$, this turns the problem of optimizing Equation (22) into the problem of optimizing a general polynomial (as x_i, x_j can be merged into a variable with a higher degree), a known NP-hard problem [17].

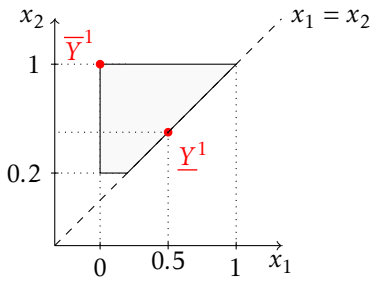


Figure 7: Multilinear optimisation with comparative queries: illustration.

5.2. Increasing (and decreasing) functions

Functions Φ that are increasing (or decreasing) in each variable are also of practical importance, as they are commonly use in applications such as reliability analysis [20], multi-criteria decision making [14], learning problems [26] such as ordinal regression [15].

When $\mathcal{X}^k$ is a hypercube (i.e., when using only local bound queries), then the computation of $\underline{Y}^k, \overline{Y}^k$ and of U_{Y^k} is straightforward, as we have when Φ is increasing

$$U_{Y^k} = \Phi(\overline{X}_1^k, \ldots, \overline{X}_n^k) - \Phi(\underline{X}_1^k, \ldots, \underline{X}_n^k), \tag{23}$$

If Φ is monotonic, but can be either increasing or decreasing in each variable x_i, computations involved in the calculation of U_{Y^k} are also significantly facilitated as the upper and lower bounds of function Φ are reached on well-known vertices of the hypercube $\mathcal{X}^k$:

$$U_{Y^k} = \Phi(\overline{X}_I^k, \underline{X}_{\overline{I}}^k) - \Phi(\underline{X}_I^k, \overline{X}_{\overline{I}}^k), \tag{24}$$

where I denotes the set of variables in which Φ is increasing, and $\overline{I}$ its complement.

When Φ is increasing, computing bounds $\underline{Y}^k, \overline{Y}^k$ when also considering comparative queries remains quite tractable. Given a set of questions, let us

first denote $\underline{X}_i^{k,LB}, \overline{X}_i^{k,LB}$ the set of bounds obtained by considering only the local bound queries made so far. using $M^{k,+}$, let us then consider the bounds

$$\underline{X}_i^k = \max_{j, M^{k,+}(j,i)=1} (\underline{X}_i^{k,LB}, \underline{X}_j^{k,LB}), \tag{25}$$

and

$$\overline{X}_i^k = \min_{j, M^{k,+}(i,j)=1} (\overline{X}_i^{k,LB}, \overline{X}_j^{k,LB}), \tag{26}$$

that are (jointly) reachable in $\mathcal{X}^k$ for each x_i, as they do not contradict any pairwise comparison. The bounds $\underline{Y}^k, \overline{Y}^k$ are then given by Equation (23). Note that this is not necessarily true if Φ is monotonic in every variable, but is increasing in some variables and decreasing in some others. An easy example is given by considering the function $\Phi(x_1, x_2) = x_1 - x_2$ together with the set of Figure 7.

Proposition 2. *Consider $\mathcal{X}^0 = \times_{i=1}^n X_i^0$, then for any comparative assessment $Q_{i,j}$, there is a local bound query $Q_i^{\alpha*}$ such that*

$$U^{mM}_{Y^k(Q_{i,j},\cdot)} \leq U^{mM}_{Y^k(Q_i^{\alpha*},\cdot)}$$

Proof. Let Φ be increasing (or decreasing) in every variable, and consider the comparative query $Q_{i,j}$, meaning that the variable intervals overlap. Since i and j are exchangeable, there are two situations to consider that are depicted in Figure 8: the case where $\underline{X}_i^0 \leq \underline{X}_j^0$ and $\overline{X}_i^0 \leq \overline{X}_j^0$ (Figure 8.a), and the case of nested intervals (Figure 8.b).

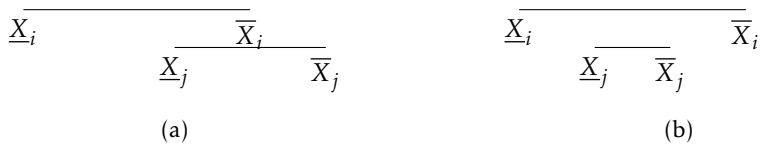


Figure 8: Overlap schemes of intervals X_i and X_j.

Let us look at these two cases in details

- case a: in this case, an answer $A = No$ do change $\underline{X}_i^0$ into $\underline{X}_i^1 = \underline{X}_j^0$ and $\overline{X}_j^0$ into $\overline{X}_j^1 = \overline{X}_i^0$, meaning that $U_{Y^0(Q_{i,j},Yes)} \leq U_{Y^0}$. However, an answer $A = Yes$ will leave both intervals $X_i^1 = X_i^0$ and $X_j^1 = X_j^0$ unchanged, meaning that $U_{Y^0(Q_{i,j},Yes)} = U_{Y^0}$ and that the minimax value of this query is

  $$U^{mM}_{Y^k(Q_{i,j},\cdot)} = \max(U_{Y^k(Q_{i,j},No)}, U_{Y(Q_{i,j},Yes)}) = U_{Y^k(Q_{i,j},Yes)} = U_{Y^{k-1}}.$$

  This means that for pairs of variables in case a, any comparative assessment cannot improve the situation according to the minimax criterion,

while local bound queries $Q_i^{\alpha^*,mM}$ or $Q_j^{\alpha^*,mM}$ can improve the situation, and are guaranteed to do so if the function is strictly increasing in every variable.

- case b: the comparative query will only update the largest interval (i.e., herein X_i): the positive answer will shift its lower bound, while the negative answer shifts its upper bound. In this situation

$$\begin{aligned} U^{mM}_{Y^0(Q_{i,j},\cdot)} &= \max(U_{Y^0(Q_{i,j},No)}, U_{Y^0(Q_{i,j},Yes)}) \\ &= \max(U_{Y^0(Q_i^{\underline{X}_j},No)}, U_{Y^0(Q_i^{\overline{X}_j},Yes)}) \\ &\geq U^{mM}_{Y^0(Q_i^\alpha)} = \max(U_{Y^0(Q_i^\alpha,No)}, U_{Y^0(Q_i^\alpha,Yes)}) \end{aligned}$$

  For any $\alpha \in [\underline{X}_j, \overline{X}_j)]$.

  Therefore, also in the case of Figure 8.b every comparative query $Q_{i,j}$ can be outperformed by a choice of $U^{mM}_{Y^0(Q_i^\alpha)}$.

In conclusion, for every pair i, j there is a local bound query at least as good as the comparative query involving i, j, hence proving Proposition 2. □

Note that Proposition 2 can be extended to the case where $\mathcal{X}^0$ (or $\mathcal{X}^k$) already contains some comparative statements. In such a case, the reduction of intervals in the proof of Proposition 2 will be propagated to other variables involved in these comparisons, and the argument still holds. This means that, in practice, comparative queries would never be selected with a maximin criterion on increasing (decreasing) functions. Yet this does not mean that comparative queries have no interest in such problems, as they can be perceived as more natural or easier to answer by experts. Hence, it may be useful to start with comparative queries to then switch to local bound queries, in order to refine the obtained results. We will also see in Section 6 that in practice comparative queries can sometimes do better than local bound ones.

5.3. Multi linear increasing functions

When Φ is both multi linear and increasing in each direction, there are interesting properties to compute our optimal strategies in local bound queries based elicitation. The first property relates to the shape of $U^{cr}_{Y^k(Q_i^\alpha,Yes)}$ and $U^{cr}_{Y^k(Q_i^\alpha,No)}$.

Proposition 3. *If Φ is a multi linear increasing function, then for every $i \in N$, $U^{cr}_{Y^k(Q_i^\alpha,Yes)}$ and $U^{cr}_{Y^k(Q_i^\alpha,No)}$ are linear in α.*

Proof. *If Φ is increasing in all variables, this means that in Equations* (12)-(13)*, the variables other than x_i in $\max \Phi(\mathbf{x})$ and $\min \Phi(\mathbf{x})$ will always be set to the same boundary values, hence $\max \Phi(\mathbf{x})$ and $\min \Phi(\mathbf{x})$ will be linear in α if Φ is multi linear.*

This has several consequences on the computations of strategies:

- Using the local bound queries, the minimax strategy recommends a unique query bound M_i in X_i, as $U^{cr}_{Y^k(Q_i^\alpha,Yes)}$ and $U^{cr}_{Y^k(Q_i^\alpha,No)}$ intersection will be a unique point;
- Using the local bound queries: computing $U^{cr}_{Y^k(Q_i^\alpha,Yes)}$ and $U^{cr}_{Y^k(Q_i^\alpha,No)}$ will require only three computations, as they are linear (requiring each two evaluations) and as they have the same maximal value. Computing M_i then comes down to evaluate the intersection point of two lines.

It should be noted that the above remarks may be true for some variables x_i (but not all, as in Proposition 3) even if the function is not increasing. This is the case in Example 3 and Figure 4 for variable x_2.

However, these properties are lost when combining both query formats. In fact, $U^{cr}_{Y^k(Q_i^\alpha,Yes)}$ and $U^{cr}_{Y^k(Q_i^\alpha,No)}$ become a polynomial function in α when there are some comparisons involving the i component, with the polynomial degree being at most the number of such comparisons. This is due to the fact that changing a lower/upper bound of some variable may also impact the lower/upper bound of other variables, as indicated by Equations (26)-(25).

6. Application to reliability

In the reliability and safety domain, several analysis such as systems comparisons, reliability prediction of newly designed systems (in early design stages), etc. suffer from the lack of empirical data. This may be due to a high cost of the data acquisition process, time constraints, complexity, etc. Experts opinions can be a valuable source of information to enhance the poor state of knowledge.

In this section, we illustrate the use of the optimal elicitation strategies described previously in reducing uncertainty in system dependability parameters. The proposed approach is particularly appealing in reliability for two reasons. First, both query formats considered in this work are natural and meaningful. In particular, comparing the performance or the behavior of the components is quite common, and experts are familar with this question format. Second, the computations are tractable given the monotonicity of the reliability functions.

6.1. Estimating the reliability of a system

In the previously considered network S, every component i is either operating or failing and its state is represented by a Boolean variable e_i that associates 0 and 1 to the failed, working state, respectively. The probability that the component is functioning is called the elementary reliability:

$$p_i = Pr(e_i = 1)$$

. When the components are independent, i.e., when their state variables are stochastically independent, the reliability of the system:

$$R = Pr(\Phi_s(e_1, ...e_n) = 1),$$

can be determined from the reliability of its components via the *reliability function* Φ:

$$R = \Phi(p_1, ..., p_n), \tag{27}$$

which is the multi-linear extension of Φ_s [20] and so writes:

$$\Phi(p_1, ..., p_n) = \sum_{A \subseteq N} d_A \prod_{i \in A} p_i$$

In practice, the exact expression of the reliability function can be directly generated using the inclusion-exclusion formula [19] based on determining the minimal path set (i.e., the minimal set of components that must be in working state that guarantees the functioning of the system) and cut sets (the minimal set of components such that if all of them fail, the system is guaranteed to fail whatever the value of the others components).

For the majority of systems, forming the class of semi-coherent systems, the structure function is monotonic in each of the n directions, meaning that we can apply the results of Section 5.3. We now consider the case where variables p_i are known to lie in intervals P_i

6.2. Case studies

We now detail two case studies. The first, focusing on local bound queries, is a simple illustrative case showing the interest of our approach compared to a random or a heuristic approach. The second, considering an actual system use in railways, compares the comparative and local bound queries, confirming experimentally what has been discussed in previous sections.

6.2.1. Local bound queries: optimal vs heuristic strategies

Let us first explore whether computing optimal answers is worth it. In order to do so, we compare the performance of the proposed method to two basic strategies, focusing on local bound queries:

1. a random strategy that compares at each stage the reliability of component i, selected at random in N, with some random $\alpha \in P_i$. For the results to be significant, the performance of the strategy at each iteration is averaged over a high number (herein 1000) of runs.
2. a baseline strategy that asks at each stage about the most uncertain component, and the query bound is the midpoint of the largest interval:

$$Q^*_{Baseline} = \left(i^*, \frac{\overline{P}_{i^*} + \underline{P}_{i^*}}{2} \right)$$

where:

$$i^* = \arg\max_{i \in N} U_{P_i}.$$

To make the comparison, we consider a simple series parallel system composed of four independent and non-identical components (Figure 9). The system reliability is:

$$R = p_1p_2p_3p_4 + p_1p_4 + p_2p_4 + p_3p_4 - p_1p_3p_4 - p_2p_3p_4 - p_1p_2p_4.$$

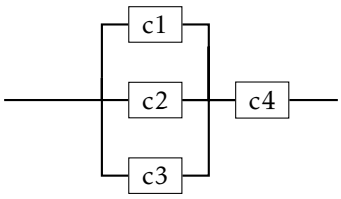


Figure 9: Reliability block diagram of a series parallel system.

Let the initial state of knowledge on the elementary reliabilities be the following: $p_1 \in [0.01, 0.99]$, $p_2 \in [0.01, 0.99]$, $p_3 \in [0.97, 0.99]$, and $p_4 \in [0.7, 0.9]$, and the true values be: $p_1^* = 0.6$, $p_2^* = 0.7$, $p_3^* = 0.99$, $p_4^* = 0.8$.

The sequence of questions and their effects on input variables $p_1, \ldots, p_4$ using the baseline strategy can be visualized in Figure 10 which plots the uncertainty on each component at every stage. A drop in the points of component i at stage $k+1$ indicates that the k^{th} query of the baseline strategy was about that component. The magnitude of the drop corresponds to the uncertainty reduction after the question has been answered. Note that up to the 6^{th} question, the strategy selected components 1 and 2, those two being the most uncertain.

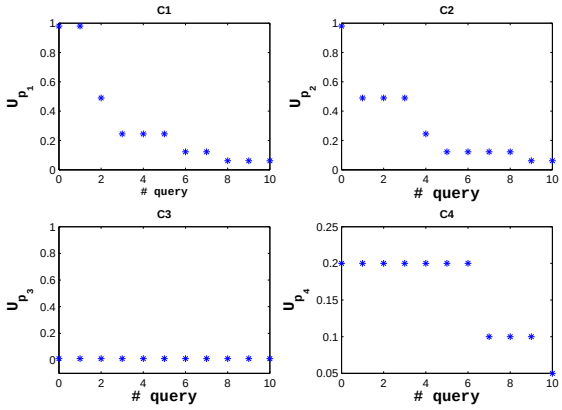


Figure 10: The baseline strategy: the sequence of optimal components.

However, reducing uncertainty on components 1 and 2 does not reduce our uncertainty on the reliability R, as shows the baseline curve in Figure 11. In this case the baseline strategy performs actually very bad, not only compared

to the optimal Bayesian and minimax strategies, but also to the random up to the 8^{th} query. This is due to the fact that the baseline basically ignores the impact that a reduction of the input uncertainty will have on the output uncertainty, or in this case how important are the component uncertainties with respect to the overall system uncertainty.

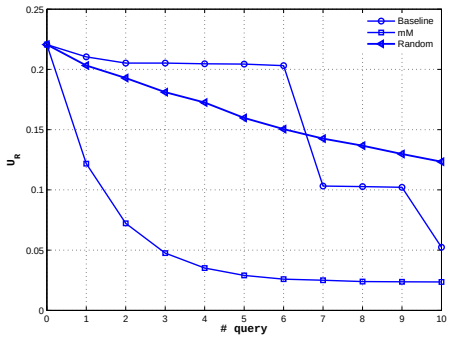


Figure 11: Minimax, random, and baseline strategies for the case of series parallel system.

This confirms that looking for optimal queries is actually useful, as it compares favourably to simple heuristic strategies in terms of number of questions asked before reaching a given precision level. This is especially true when input variables or components do not have symmetric roles in the output uncertainty.

6.2.2. Local bound and comparative queries: the RBC system

We now focus on comparing the two types of queries explored in detail in the paper, and to so consider a real-world system used in the European railways traffic management system: the Radio Block Center system (RBC), whose role is to collect data about the position of trains and to provide movement authorisation. Because of the relatively recent exploitation of the system, sufficient data to estimate the reliability of the RBC are lacking, so it is a perfect candidate for our framework.

The RBC is composed of 5 different components, each of them being redundant. The architecture of the RBC is pictured in Figure 12, where the 2/3 symbol means that the subsystem composed of components 5 works if and only if at least 2 out of the 3 components work.

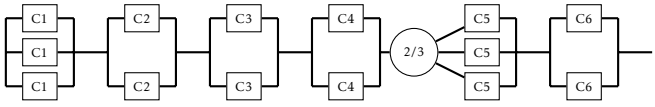


Figure 12: Reliability block diagram of the RBC.

The reliability function can be written as:

$$R = (1-(1-p_1)^3)(1-(1-p_2)^2)(1-(1-p_3)^2)(1-(1-p_4)^2)p_{tmr}$$

with

$$p_{tmr} = (3p_5^2) - 2(p_5^3))(1-(1-p_6)^2).$$

We consider a setting where initial knowledge tells us that the reliabilities of the RBC components range respectively in $[0.3, 1]$, $[0.2, 0.8]$, $[0.3, 1]$, $[0.6, 0.7]$, $[0.3, 1]$, $[0.45, 0.55]$, and that the true values are: $p_1^* = 0.45$, $p_2^* = 0.7$, $p_3^* = 0.6$, $p_4^* = 0.68$, $p_5^* = 0.8$, $p_6^* = 0.5$. Figure 13 shows the uncertainty reduction profiles (for the minimax and Bayesian strategies) using solely comparative queries, while Figure 14 also shows the profiles (for the minimax and Bayesian strategies) using solely local bound queries. Proposition 2 indicates that considering both types of queries would be equivalent to consider solely local bound queries. About these figures, we can notice that:

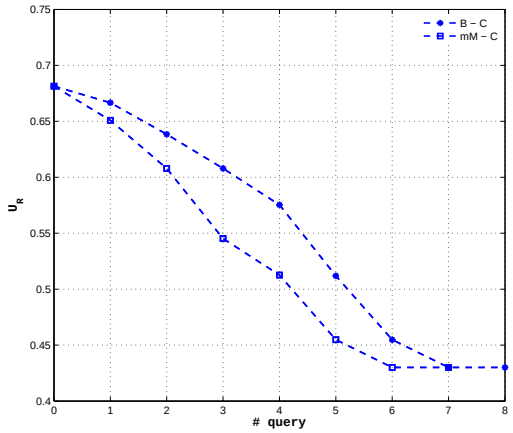


Figure 13: Performance of the pairwise comparisons-based elicitation.

- in both cases, the minimax strategy actually performs better than the Bayesian one. This shows that in some cases, taking a comparative approach can actually lead to better results;
- the limitations of using comparative queries, compared to local bound queries, is quite visible on Figure 14. Indeed, with comparative queries at most an uncertainty reduction of 37% of the initial one. Although we cannot expect qualitative information (in the form of simple comparisons) to outperform quantitative ones, comparative queries can still bring some interesting information, and by being simpler can also be used at the start of the elicitation procedure, as they usually require less cognitive efforts;

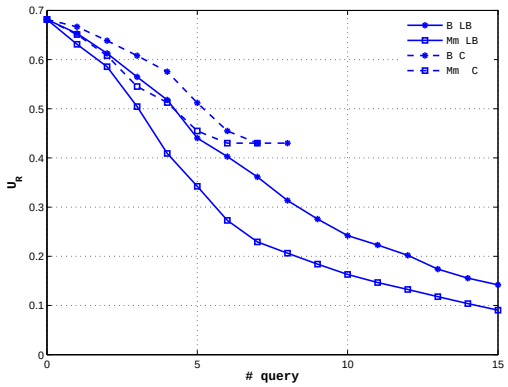


Figure 14: Profile of uncertainty reduction induced by comparative (C) and local bound questions (LB) for both the minimax (mM) and probabilistic (B) strategies.

- for local bound queries, the minimax strategy generally requires in average three less queries than the Bayesian one to reach a similar uncertainty reduction. This is a noticeable gain, showing that the two strategies can lead to very different results and are not equivalent.

We said that comparative queries could be useful as a "warm-up", as they are easier to answer. However, the next situation shows that starting by them could also lead to better (i.e., less queries for a similar uncertainty reduction) results. Consider now another scenario where the initial state of knowledge is the following: $p_1 = [0.4, 0.5]$, $p_2 = [0.61, 0.62]$, $p_3 = [0.45, 0.85]$, $p_4 = 0.45$, $p_5 = [0.9, 1]$, $p_6 = [0.46, 1]$, $p_1 \leq p_3$ and $p_3 \leq p_6$. The true values are: $p_1^* = 0.45$, $p_2^* = 0.615$, $p_3^* = 0.63$, $p_4^* = 0.45$, $p_5^* = 0.95$, $p_6^* = 0.7$.

We consider two strategies: a first one where we ask first all possible comparative queries (simpler and easier to answer) then switch to local bound ones, and a second where just consider local bound queries. Figure 13 shows the resulting profile of uncertainty reduction, as a continuous line for the first strategy, and as a dashed line for the second one.

A quite noticeable result is that the first comparative query ($Q_{2,3}$) actually outperform in practice the first local bound query ($Q_3^{0.57}$). However, the two next comparative queries do not improve much the situation, and in particular are outperformed by the corresponding local bound queries. This means that if we had asked only one comparative query to start and then had switch to local bound queries, we would have obtained a better strategy than those displayed in Figure 13. This tells us that comparative queries, used in limited number, may not only be interesting as easy or warm-up questions, but may actually lead to more efficient elicitation procedure. Also remember that this case study involves 5 components and that in the scenario only 3 comparative queries are performed, hence asking all 3 of them was likely to be sub-optimal.

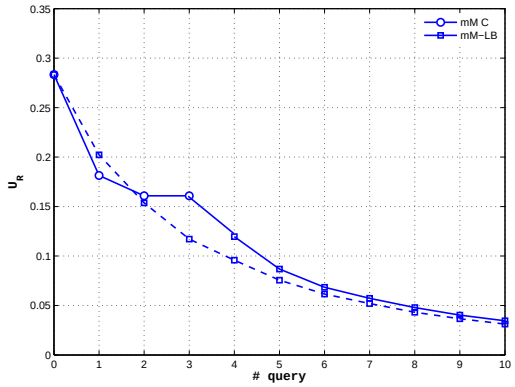


Figure 15: Profile of uncertainty reduction induced by comparative (C) and local bound questions (LB) for both the minimax (mM) and probabilistic (B) strategies.

7. Conclusion

In this paper, we have explored the problem of eliciting expert information in an optimal way to reduce set uncertainty, partly taking inspiration from recent works in preference learning and elicitation. This is done by considering questions easily understandable by experts, and by defining a way to compare the values of these different questions.

While the general framework introduced may face computational challenges, we have investigated cases of practical interest in which it remains quite tractable. We have also provided some first results and applications on reliability assessment issues.

This work in can be enriched, complemented, and extended in several ways. Beyond the obvious ones of investigating other queries and types of questions, we think two challenging avenues of research are particularly promising:

- adapt the current criteria and define efficient strategy search algorithms for the simultaneous and sequential versions of the problem (see Figure 3). Here again, preference elicitation could be a starting point [5];
- combine the current approach, focusing on expert, with experimental design techniques [13]. Indeed, one could consider an experiment (to be performed) as a question, and its result as an answer impacting our knowledge of the system. Expertise and experimental data points could then be seen under the same umbrella.

Acknowledgments

This work has been supported by the ANR project RECIF (ANR-13-JS03-0007).

References

[1] Ambati, V., Vogel, S., and Carbonell, J. (2010). Active learning-based elicitation for semi-supervised word alignment. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 365–370. Association for Computational Linguistics.

[2] Aspinall, W. and Cooke, R. (2013). Quantifying scientific uncertainty from expert judgement elicitation. *Risk and Uncertainty Assessment for Natural Hazards*, page 64.

[3] Ben Abdallah, N. and Destercke, S. (2015). Optimal expert elicitation to reduce interval uncertainty. In press, A., editor, *Uncertainty in Artificial Intelligence*, Uncertainty In Artificial Intelligence (UAI) 2015, pages 12–22, Amsterdam, Netherlands.

[4] Benabbou, N., Perny, P., and Viappiani, P. (2014). Incremental Elicitation of Choquet Capacities for Multicriteria Decision Making. In *Proceedings of ECAI'14*, pages 87–92.

[5] Boutilier, C. (2002). A pomdp formulation of preference elicitation problems. *Proceedings of AAAI Conference*, 1(3):321–323.

[6] Boutilier, C., Filmus, Y., and Oren, J. (2013). Efficient vote elicitation under candidate uncertainty. IJCAI.

[7] Boutilier, C., Patrascu, R., Poupart, P., and Schuurmans, D. (2006). Constraint-based optimization and utility elicitation using the minimax decision criterion. *Artificial Intelligence*, 170(8-9):686–713.

[8] Brightwell, G. and Winkler, P. (1991). Counting linear extensions. *Order*, 8(3):225–242.

[9] Curtis, A. and Wood, R. (2004). Optimal elicitation of probabilistic information from experts. *Geological Society, London, Special Publications*, 239(1):127–145.

[10] Drenick, R. (1992). Multilinear programming: Duality theories. *Journal of optimization theory and applications*, 72(3):459–486.

[11] Drenick, R. (1994). Multilinear programming: duality theories. *Journal of optimization theory and applications*, 81(2):421–422.

[12] Dyer, M. and Frieze, A. (1988). On the complexity of computing the volume of a polyhedron. *SIAM Journal on Computing*, 17(5):967–975.

[13] Fedorov, V. (2010). Optimal experimental design. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(5):581–589.

[14] Grabisch, M. and Labreuche, C. (2003). On the extension of pseudo-boolean functions for the aggregation of interacting criteria. *European Journal of Operational Research*.

[15] Greco, S., Mousseau, V., and 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.

[16] Hammer, P. and Rudeanu, S. (1968). *Boolean Methods in Operations Research and Related Areas*. Springer, Berlin Heidelberg, New York.

[17] Kreinovich, V., Lakeyev, A., and Rohn, J. (1996). Computational complexity of interval algebraic problems: Some are feasible and some are computationally intractable-a survey. *Mathematical Research*, 90:293–306.

[18] Laneve, C., Luscu, T., and Vania, S. (2010). The interval analysis of multilinear expressions. *Electronic Notes in Theroretical Computer Science*, 267(2):43–53.

[19] Lin, P., Leon, B., and Huang, T. (1976). A new algorithm for symbolic system reliability analysis. *Reliability, IEEE Transactions on*, R-25(1):2–15.

[20] Marichal, J.-L. (2014). Structure functions and minimal path sets. *arXiv preprint arXiv:1401.0803*.

[21] McCormick, G. (1976). Computability of global solutions to factorable nonconvex programs: Part i - convex underestimating problems. *Mathematical Programming*, 10:147–175.

[22] Owen, G. (1972). Multilinear extensions of games. *Management Sciences*, 18:64–79.

[23] Rikun, A. (1997). A convex envelope formula for multilinear functions. *Journal of Global Optimization*, 10(4):425–437.

[24] Settles, B. (2012). Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114.

[25] Sherali, H. (1997). Convex envelopes of multilinear functions over a unit hypercube and over special discrete sets. *Acta Mathematica Vietnamica*, 22(1):245–270.

[26] Tehrani, A. F., Cheng, W., Dembczyński, K., and Hüllermeier, E. (2012). Learning monotone nonlinear models using the choquet integral. *Machine Learning*, 89(1-2):183–211.

[27] Viappiani, P. and Kroer, C. (2013). Robuts optimization of recommendation sets with the maximin utility criterion. *Proceedings of the Third Algorithmic Decision Theory International Conference*, pages 411–424.

[28] Wald, A. (1945). Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, pages 265–280.

[29] Wang, T. and Boutilier, C. (2003). Incremental utility elicitation with the minimax regret decision criterion. In *IJCAI*, pages 309–318.

[30] Warshall, S. (1962). A theorem on boolean matrices. *Journal of the ACM (JACM)*, 9(1):11–12.

[31] Yannakakis, M. (1991). Expressing combinatorial optimization problems by linear programs. *Journal of Computer and System Sciences*, 43(3):441 – 466.