Characterizing coherence, correcting incoherence

Erik Quaeghebeur[1,2,*]

Abstract

Lower previsions defined on a finite set of gambles can be looked at as points in a finite-dimensional real vector space. Within that vector space, the sets of sure loss avoiding and coherent lower previsions form convex polyhedra. We present procedures for obtaining characterizations of these polyhedra in terms of a minimal, finite number of linear constraints. As compared to the previously known procedure, these procedures are more efficient and much more straightforward. Next, we take a look at a procedure for correcting incoherent lower previsions based on pointwise dominance. This procedure can be formulated as a multi-objective linear program, and the availability of the finite characterizations provide an avenue for making these programs computationally feasible.

Keywords: Coherence, avoiding sure loss, polytope theory, multi-objective linear programming, incoherence, dominance

1. Introduction

In the theory of coherent lower previsions (for an overview, see Walley's book [1] or Miranda's survey [2]), its coherence condition takes a central role: it defines which models—lower previsions—are fully rational, meaning that they do not implicitly encode commitments—in terms of buying prices for gambles—that are more demanding than the ones explicitly made. The consequences of this criterion have been extensively studied both in the unconditional and the conditional case, in finite and infinite spaces.

In this paper, we study the coherence criterion for unconditional lower previsions defined on a finite set of gambles, which in turn are essentially defined on a finite possibility space. What can we still add in this restricted setting? Results that make new numerical applications feasible, namely, procedures for obtaining a characterization of coherence in terms of a minimal, finite number of linear constraints that are more efficient than the existing one. These results are presented in Section 4. Note that our procedures give an answer to the question "Which lower previsions are coherent?", and should not be confused with verification procedures, which deal with the question "Is this specific lower prevision coherent?". Of course, the characterization our procedures generate can be used for verification purposes, but this may be reasonable only if many verifications need to be performed

One may wonder what new kinds of applications are possible once we have a minimal linear constraints characterization? In Section 5, we provide one example in a proposal for a method to correct an incoherent lower prevision downward to make it coherent. Similarly to natural extension, this method is formulated in terms of pointwise dominance of lower previsions.

Because of the finitary context of this paper and its aim to be an enabler for numerical applications, it is advantageous to reformulate a number of variants of the coherence criterion and the related criterion of avoiding sure loss in matrix terms; we do this in Section 3.

*Corresponding author

Email address: Erik.Quaeghebeur@cwi.nl (Erik Quaeghebeur)

[1]Currently at the Centrum Wiskunde & Informatica. Postal address: Postbus 94079, 1090 GB Amsterdam, The Netherlands.

[2]This paper was written partially when the author was working for the SYSTeMS Research Group of Ghent University, partially as guest researcher of the Department of Information and Computing Sciences of Utrecht University, and partially as guest researcher at the Dutch Centrum Wiskunde & Informatica (CWI). Revision of this work was carried out while Erik Quaeghebeur was an ERCIM "Alain Bensoussan" Fellow, a program receiving funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 246016.

Published in Elsevier's International Journal of Approximate Reasoning: doi:10.1016/j.ijar.2014.03.005

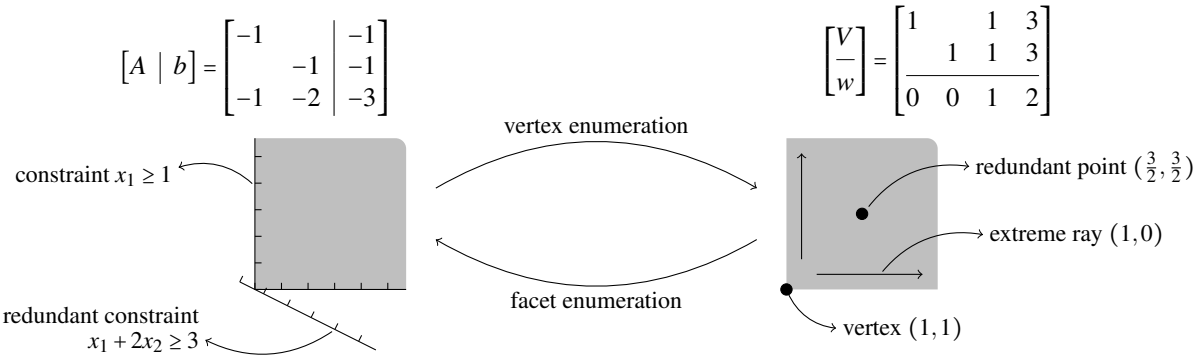

Figure 1: Illustration of basic polytope theory concepts using a toy two-dimensional example polyhedron. The H-representation is made explicit on the left and the V-representation is made explicit on the right. For both the matrix representation (including redundancy) is given on top, and below a graphical illustration is given, with the polyhedron itself in gray.

We make use of polytope theory concepts throughout this paper. We also make use of multi-objective linear programming both for our downward correction method as well as for some of our procedures to obtain the minimal linear constraints characterization. Therefore, we start out with short primers on these topics in Section 2.

2. Primers

2.1. Polytope theory essentials

Let us review some concepts and techniques from polytope theory; for more information, consult, e.g., Grünbaum [3], Ziegler [4], or Fukuda [5]. Any convex polyhedron in a n-dimensional space can be described in two ways:

As an *H-representation* $\{x \in \mathbb{R}^n : Ax \leq b\}$: A set of k linear constraints (inequalities/half-spaces) defined by a matrix A in $\mathbb{R}^{k \times n}$ and a column vector b in $\mathbb{R}^k$; denoted compactly as $[A, b]$, where the comma denotes horizontal concatenation of matrices.

As a *V-representation* $\{V\mu : \mu \in \mathbb{R}^\ell \wedge \mu \geq 0 \wedge w\mu = 1\}$: A set of ℓ points and rays, defined by a matrix V in $\mathbb{R}^{n \times \ell}$ and a row vector w in $(\mathbb{R}_{\geq 0})^\ell$, with the zero components indicating rays; denoted compactly as $[V; w]$, where the semicolon denotes vertical concatenation of matrices.

The two representations are dual in the sense that $[A^\top; b^\top]$ is the V-representation of some polyhedron and $[V^\top, w^\top]$ is the H-representation of some—possibly different—polyhedron. This duality is also present in the algorithms of polytope theory.

H- and V-representations may contain redundant constraints and points or rays, i.e., those that are implied by the other constraints or the other points or rays. Non-redundant points and rays are called vertices (or extreme points) and extreme rays. Let i be the total number of constraints or points and j the non-redundant number; redundancy removal algorithms essentially require solving i linear programming problems of size $n \times j$ [6].

Moving between the H- and V-representations is done using vertex enumeration algorithms and the dual facet enumeration algorithms. There are enumeration algorithms with a complexity linear in n, k, and ℓ [7]. Nevertheless, enumeration is inherently highly complex, as ℓ can be exponential in k and vice versa.

Projecting a polyhedron is straightforward in V-representation: project the vertices and then remove the redundant ones. However, in H-representation the best technique depends on the polyhedron's properties: the classical approach, Fourier–Motzkin elimination, is inefficient and on top of that generates a lot of redundant constraints; another approach, block elimination, is inefficient when the number of vertices is high, which is common. The equality set projection approach is claimed to be useful in such cases [8], but our input data caused errors in the available code [9].

The concepts introduced in this section are illustrated in Figure 1.

Below, we assume that the output of enumeration and projection algorithms is minimal, i.e., non-redundant.

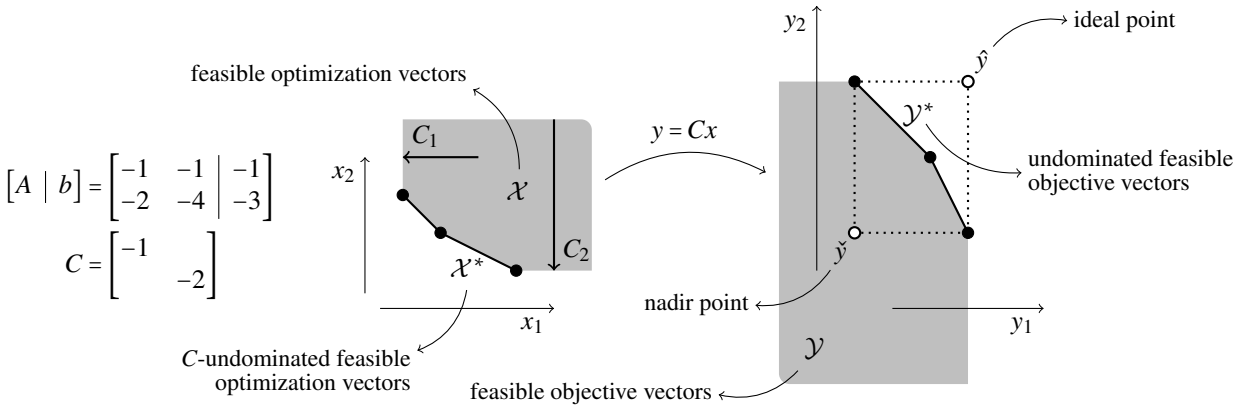

Figure 2: Illustration of multi-objective linear programming, with $n = m = 2$. On the far left, the matrices $[A, b]$ and C defining the problem are given. To their right, we give a graphical illustration of the problem: The sets $\mathcal{X}$ and $\mathcal{Y}$ are shaded gray. The sets $\mathcal{X}^*$ and $\mathcal{Y}^*$ are shown as black lines. The members of $\operatorname{ext}\mathcal{X}^*$ and $\operatorname{ext}\mathcal{Y}^*$ are shown as black dots. The vectors C_1 and C_2—rows of C—that point towards higher objective vector component values are drawn free: only their direction and magnitude matter. The ideal point and nadir point are given as white-filled dots.

2.2. Multi-objective linear programming

We here give a brief introduction to multi-objective linear programming; for more information, see, e.g., Ehrgott [10]. We assume familiarity with standard, single objective linear programming; if not, have a quick look at a standard reference such as Bertsimas and Tsitsiklis's book [11].

Any *multi-objective linear program* can be put in the following form:

$$\begin{aligned} &\text{maximize} \quad y = Cx, \\ &\text{subject to} \quad Ax \leq b \ \text{ and } \ x \geq 0. \end{aligned} \tag{1}$$

In this program, x denotes the n-dimensional real *optimization vector*, y is the m-dimensional *objective vector*, and $Ax \leq b$ is a set of k *linear constraints*; so we assume $C \in \mathbb{R}^{m \times n}$, $A \in \mathbb{R}^{k \times n}$, and $b \in \mathbb{R}^k$ as given. Vector inequalities should be read as follows: $x \geq z \Leftrightarrow \min(x - z) \geq 0$ and $x > z \Leftrightarrow x \geq z \wedge x \neq z$. Here, min (max) selects its argument vector's minimum (maximum) component value.

Whereas in single objective linear programming, with $m = 1$, all optimization vectors x are completely ordered by the single objective, whenever $m > 1$, they are only partially ordered through the standard ordering of the objective vectors. Consequently, whereas in single objective linear programming all optimal solutions are equivalent from the objective value point of view, in multi-objective linear programming there are in general multiple sets of incomparable 'Pareto' optimal (or 'efficient')—i.e. *C-undominated*—solutions.

The sets of feasible optimization and objective vectors are $\mathcal{X} \coloneqq \{x \in \mathbb{R}^n : Ax \leq b \wedge x \geq 0\}$ and $\mathcal{Y} \coloneqq \{Cx : x \in \mathcal{X}\}$, respectively. Furthermore, $\mathcal{X}^* \coloneqq \{x \in \mathcal{X} : (\forall z \in \mathcal{X} : Cx \not\leq Cz)\}$ is the set of C-undominated solutions, and so $\mathcal{Y}^* \coloneqq \{Cx : x \in \mathcal{X}^*\}$ is the set of undominated objective vectors. The sets of extreme points of the sets of undominated solutions and objectives are $\operatorname{ext}\mathcal{X}^*$ and $\operatorname{ext}\mathcal{Y}^*$, respectively.

The upper and lower envelopes of $\mathcal{Y}^*$ are respectively called the *ideal point* $\hat{y}$ and the *nadir point* $\check{y}$. So they provide bounds on the values attained by the undominated objective vector components. Their i-components are defined by $\hat{y}_i = \max\{y_i : y \in \mathcal{Y}^*\} = \max\{y_i : y \in \mathcal{Y}\}$ and $\check{y}_i = \min\{y_i : y \in \mathcal{Y}^*\}$. So we see that the components of the ideal point can be calculated by solving a linear program, whereas we cannot in general do this for the nadir point components.

The concepts introduced in this section are clarified in Figure 2.

The main computational tasks are, in non-decreasing order of complexity:

M1. Finding the ideal point $\hat{y}$, which can be done by solving a linear program maximizing each of the components of y separately.

M2. Finding the nadir point $\check{y}$; for algorithms, see the papers by Ehrgott and Tenfelde-Podehl [12] and Alves and Costa [13].

M3. Finding the extreme points $\operatorname{ext}\mathcal{Y}^*$ and the whole set $\mathcal{Y}^*$ of undominated objective vectors; for algorithms, which are relatively efficient only if m is small compared to n, see the papers by Benson [14] and Ehrgott et al. [15].

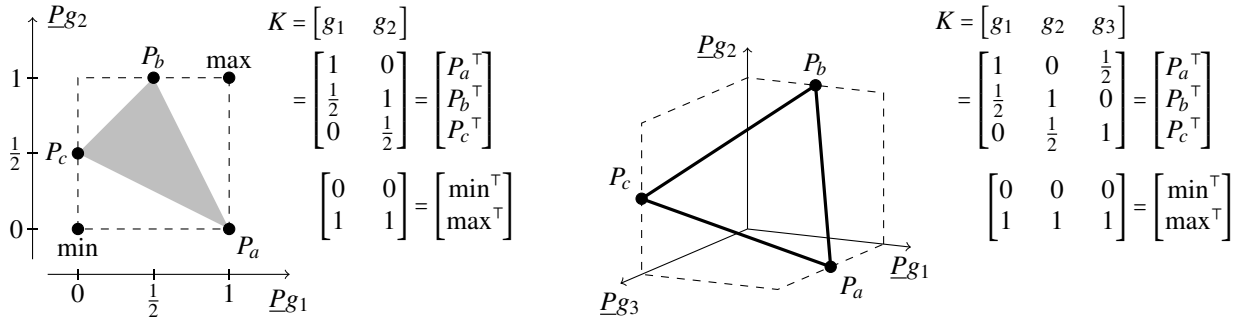

Figure 3: Illustration of sets of linear previsions for $\Omega = \{a, b, c\}$. On the left, we give an example with two gambles. On the right, we give one with three gambles. For both, we give the gamble matrix K used and its relationship to the gambles g_i ($i \in \{1, 2, 3\}$) and the degenerate previsions P_ω ($\omega \in \Omega$). In the drawings, the axes graphically represent the canonical basis of lower prevision space. The dashed lines indicate the (two and three-dimensional) intervals defined by the gamble value bounds; these are a rectangle and a cuboid, respectively. The degenerate previsions are indicated with a black dot; their convex hull, the set of linear previsions, is shaded gray in the two-gamble case and is delimited by black lines in the three-gamble case. Also given are the 'min' and 'max' lower previsions; these are also indicated in the drawing on the left for the two-gamble case.

M4. Finding the extreme points $\operatorname{ext}\mathcal{X}^*$ of the set of optimal optimization vectors; for algorithms, MOLP simplex solvers, see, e.g., the work of Evans and Steuer [16], Strijbosch et al. [17], or Ehrgott [10, Sec. 7].

M5. Finding the whole set $\mathcal{X}^*$ of optimal optimization vectors; for algorithms, based on post-processing the MOLP simplex solver output, see, e.g., the papers by Yu and Zeleny [18] or Isermann [19].

3. Matrix formulations of avoiding sure loss and coherence

Consider a finite possibility space Ω and a finite set of gambles $\mathcal{K} \subset \mathbb{R}^\Omega$ on this possibility space. Each gamble g in $\mathcal{K}$ can be looked at as a vector: some ordering of the possibility space Ω is chosen and if ω is the jth element of Ω in the order, then $g\omega$ is the jth component of the vector corresponding to g. We group these gamble vectors, also ordered in some way, as columns in a gamble matrix $K \in \mathbb{R}^{\Omega \times \mathcal{K}}$. We use the same notation for scalars and constant vectors; the identity matrix is denoted $\mathbb{I}$; there will be no ambiguity in this paper because we leave their size implicit.

The columns of $K^\top$ (rows of K) correspond to the degenerate previsions; explicitly: let ω be the jth element of Ω, then the jth column of $K^\top$ has as components $g\omega$ for g in $\mathcal{K}$, so it corresponds to the degenerate prevision that places all probability mass on ω. So their convex hull $\{K^\top\mu : \mu \geq 0 \wedge 1^\top\mu = 1\}$ is the set of linear previsions. Similarly, any lower prevision $\underline{P}$ defined on $\mathcal{K}$ can be looked at as a (column) vector in $\mathbb{R}^\mathcal{K}$: if g is the ith gamble in $\mathcal{K}$, then $\underline{P}g$ is the ith component of this vector. Similarly, min and max can also thought of at as (column) vectors in $\mathbb{R}^\mathcal{K}$; their components are, respectively, the minima and maxima of the columns of K—or, of the gambles in $\mathcal{K}$.

The view of lower prevision space described above is illustrated in Figure 3.

3.1. Avoiding sure loss

A lower prevision $\underline{P}$ on $\mathcal{K}$ is said to *avoid sure loss* [1, §2.4] if and only if

$$\forall \lambda \geq 0 : \ \underline{P}^\top\lambda 1 \leq K\lambda, \tag{2}$$

or, scalarized, if

$$\forall \lambda \geq 0 : \ \underline{P}^\top\lambda \leq \max(K\lambda), \tag{3}$$

or, as a linear program [20, §2.4], if

$$\nexists \lambda \geq 0 : \ (K - 1\underline{P}^\top)\lambda \leq -1, \tag{4}$$

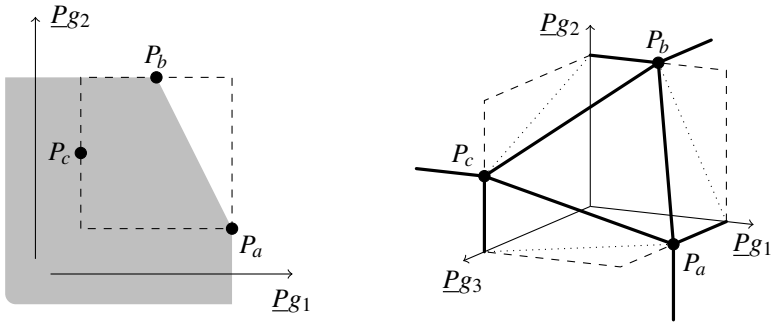

Figure 4: The polyhedra of lower previsions that avoid sure loss for the gamble sets from Figure 3. On the left, the polyhedron for the two-gamble case is given shaded in gray; on the right, the polyhedron for the three-gamble case is is given delimited by black lines. Note that, despite the fact that the shading and black lines are bounded, these polyhedra are unbounded in negative lower prevision value directions. We can see that these polyhedra consist of all lower previsions dominated by some linear prevision by comparing this figure with Figure 3.

or, based on dominance by a linear prevision [1, §3.3.3(a)], if

$$\exists \mu \geq 0 : \ \underline{P} \leq K^{\top}\mu \ \ \wedge \ \ 1^{\top}\mu = 1, \tag{5}$$

or, by introducing slack variables, if

$$\exists \mu, \nu \geq 0 : \ \underline{P} = K^{\top}\mu - \mathbb{I}\nu \ \ \wedge \ \ 1^{\top}\mu = 1. \tag{6}$$

This last form shows that the set of all sure loss avoiding lower previsions is a convex polyhedron by providing a V-representation

$$\left[\frac{V}{w}\right] \coloneqq \left[\frac{K^{\top} \quad -\mathbb{I}}{1^{\top} \quad 0^{\top}}\right]. \tag{7}$$

For the examples given in Figure 3, we have visualized this polyhedron in Figure 4.

3.2. Coherence

Let $\mathcal{S}$ denote the set of matrices obtained from the identity matrix by changing at most one 1 to -1. Then a lower prevision $\underline{P}$ on $\mathcal{K}$ is called *coherent* [1, §2.5] if and only if

$$\forall S \in \mathcal{S} : \forall \lambda \geq 0 : \ \underline{P}^{\top} S\lambda 1 \leq KS\lambda, \tag{8}$$

or, scalarized, if

$$\forall S \in \mathcal{S} : \forall \lambda \geq 0 : \ \underline{P}^{\top} S\lambda \leq \max(KS\lambda), \tag{9}$$

or, by formal analogy to Equation (4), if

$$\forall S \in \mathcal{S} : \nexists \lambda_S \geq 0 : \ (K - 1\underline{P}^{\top})S\lambda_S \leq -1, \tag{10}$$

or, by formal analogy to Equation (5) and because $S^{\top} = S$, if

$$\forall S \in \mathcal{S} : \exists \mu_S \geq 0 : \ S\underline{P} \leq SK^{\top}\mu_S \ \ \wedge \ \ 1^{\top}\mu_S = 1, \tag{11}$$

or, by introducing slack variables and because $S^{-1} = S$, if

$$\forall S \in \mathcal{S} : \exists \mu_S, \nu_S \geq 0 : \ \underline{P} = K^{\top}\mu_S - S\nu_S \ \ \wedge \ \ 1^{\top}\mu_S = 1. \tag{12}$$

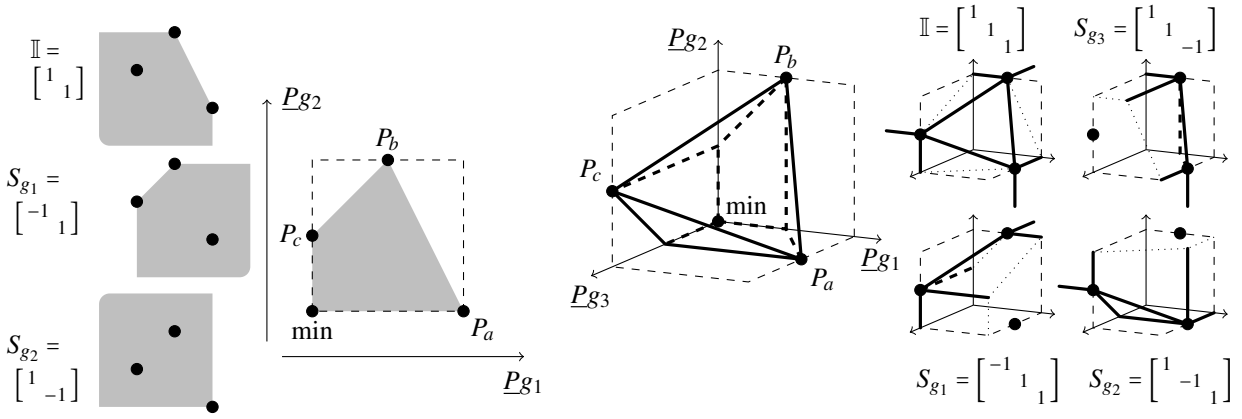

Figure 5: The polytopes of coherent lower previsions for the gamble sets from Figure 3. On the middle left, the polytope for the two-gamble case is given shaded in gray; on the middle right, the polytope for the three-gamble case is is given delimited by black lines. To show how these polytopes arise as the intersection of the polyhedra for avoiding sure S-loss, we have drawn these $|\mathcal{S}|$ polyhedra—smaller—on the outside of both polytopes, each time accompanied by the matrix S they correspond to.

This last form shows that the set of all coherent lower previsions is an intersection of $|\mathcal{S}| = |\mathcal{K}| + 1$ convex polyhedra, each containing the lower previsions that *avoid sure S-loss*—i.e., that are S-dominated by a linear prevision—, with V-representations

$$\left[\frac{V_S}{w_S}\right] \coloneqq \left[\frac{K^\top \quad -S}{1^\top \quad 0^\top}\right]. \tag{13}$$

It therefore is a convex polyhedron. Furthermore, coherence implies that $\min \leq \underline{P} \leq \max$ [1, §2.6.1(a)], so the set of coherent lower previsions is a bounded polyhedron, i.e., a *polytope*. For the examples given in Figure 3, we have visualized this polytope in Figure 5.

3.3. Further considerations

We will later on in this paper use the Lower Envelope Theorem [1, §2.6.3]:

Theorem 1. *The lower envelope $\underline{P}$ of a subset $\mathcal{Q}$ of the coherent lower previsions on a set of gambles $\mathcal{K}$ is coherent. (So $\underline{P}f \coloneqq \inf_{\underline{Q} \in \mathcal{Q}} \underline{Q}f$ for each gamble f in $\mathcal{K}$.)*

We give a proof based on Equation (11)—a version of the coherence criterion a search of ours left unencountered in the literature:

Proof. By coherence of the $\underline{Q}$ in $\mathcal{Q}$, we have a vector $\mu_{\underline{Q},S}$ such that $S\underline{Q} \leq SK^\top \mu_{\underline{Q},S}$ for each S in $\mathcal{S}$. By the lower envelope definition, for $S \coloneqq \mathbb{I}$, we have $\underline{P} \leq \underline{Q} \leq K^\top \mu_{\underline{Q},\mathbb{I}}$ for any $\underline{Q}$ in $\mathcal{Q}$. For other S, let g_S denote the gamble corresponding to the -1 diagonal component in S. Let $\underline{Q}_S$ be a coherent lower prevision from $\mathcal{Q}$ such that $\underline{P}g_S = \underline{Q}_S g_S$. Then $S\underline{P} \leq S\underline{Q}_S \leq SK^\top \mu_{\underline{Q}_S,S}$. □

In the sizable literature on verification procedures—which are typically formulated in the more general conditional context—, there is a clear separation between algorithms based on criteria formulations of the type of Equations (3) and (9) such as those by Walley et al. [20], and those of the type of Equations (5)–(6) and (11)–(12) such as those by Vicig [21] and Biazzo and Gilio [22]. This separation is also present in the characterization procedures we present; the latter type leads to the procedures in Section 4.1, the former to those in Section 4.2.

4. Computing constraints efficiently

Building on earlier work with lower probabilities [1, 23, 24], we presented a procedure for obtaining characterizations of the polytope of coherent lower previsions in terms of a minimal, finite number of linear constraints [25].

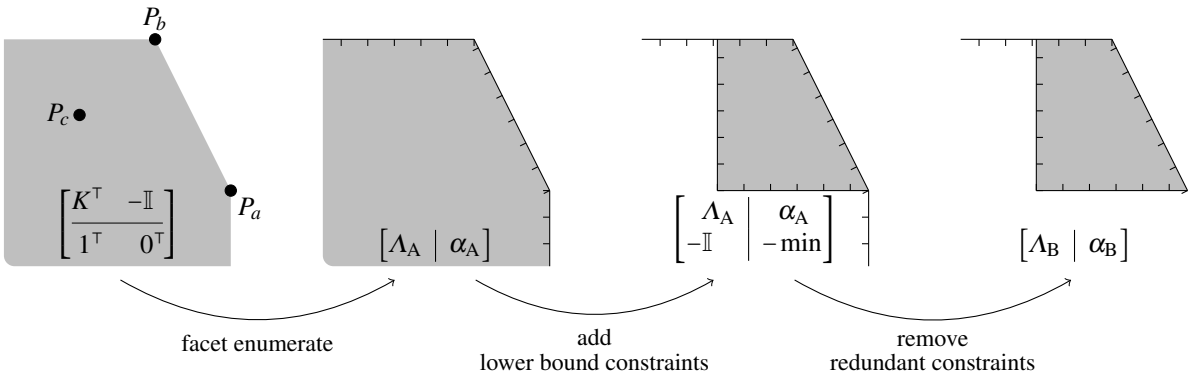

Figure 6: Illustration of Procedures A1 and B1 for the two-gamble example introduced in Figure 3, starting from the drawing of the polyhedron of lower previsions that avoid sure loss first introduced in Figure 4. In this illustration, the first step corresponds to Procedure A1 and the second and third steps correspond to the extra work done in Procedure B1. Bestubbled lines indicate constraints: the stubbles point to the half-space the constraint corresponds to.

However, the procedure is such that a relatively large number of redundant constraints are generated, which at a later step need to be removed—a computationally demanding task. Moreover, the procedure and its derivation is somewhat involved.

It is possible to derive procedures in a more direct way. Some of these more direct procedures turn out to be computationally more efficient as well, resulting in running times that are up to an order of magnitude shorter.

What are our concrete goals? We wish to find minimal H-representations for the set of all lower previsions $\underline{P}$

A. that avoid sure loss ($[\Lambda_\mathrm{A}, \alpha_\mathrm{A}]$),
B. that avoid sure loss and for which $\underline{P} \geq \min$ ($[\Lambda_\mathrm{B}, \alpha_\mathrm{B}]$),
C. that are coherent ($[\Lambda_\mathrm{C}, \alpha_\mathrm{C}]$).

So for each goal, we want to obtain a block matrix $[\Lambda, \alpha]$ that stands for the linear constraints $\Lambda \underline{P} \leq \alpha$.

These goals are formulated based on experimental results from earlier work [23–25]: For coherence, we observed that the V-representations have a much larger size than the H-representations, and to such a degree that it currently seems impractical to generate and use them. We observed that avoiding sure loss with lower bound constraints leads to a smaller H-representation than plain avoiding sure loss. As the lower bound constraints are uncontroversial in most contexts, it may be useful to use this combination as a 'lighter' proxy for plain avoiding sure loss.

Below, we first discuss the direct procedures and follow this up with a look at improved versions of our earlier, involved approach. We close the section with a short discussion of our numerical experiments.

4.1. Straightforward procedures

The straightforward procedures for Goal A go as follows:

A1. Apply a facet enumeration algorithm to the V-representation of the polyhedron of lower previsions that avoid sure loss in Equation (7) to obtain $[\Lambda_\mathrm{A}, \alpha_\mathrm{A}]$. An illustration of this procedure is given in Figure 6.

A2. As can be seen from Equation (5), we know an H-representation for pairs $[\underline{P}; \mu_\mathbb{I}]$ of which the $\underline{P}$-components are lower previsions that avoid sure loss:

$$\left[A_{\mathbb{I},\underline{P}} \quad A_{\mathbb{I},\mu_\mathbb{I}} \mid b_0\right] \coloneqq \left[\begin{array}{cc|c} \mathbb{I} & -K^\top & 0 \\ & -\mathbb{I} & 0 \\ & 1^\top & 1 \\ & -1^\top & -1 \end{array}\right]. \tag{14}$$

Project this H-representation onto the $\underline{P}$-part to obtain $[\Lambda_\mathrm{A}, \alpha_\mathrm{A}]$.

The straightforward procedures for Goal B build on those for Goal A:

B1. Start from the H-representation $[\Lambda_\mathrm{A}, \alpha_\mathrm{A}]$ resulting from Procedures A1 and A2. Then add the lower bound constraints to it, i.e., the block row $[-\mathbb{I}, -\min]$, where min denotes the column vector of gamble minima, i.e.,

minima of the columns of K. This results in the H-representation

$$\left[\begin{array}{c|c} \Lambda_{\mathrm{A}} & \alpha_{\mathrm{A}} \\ -\mathbb{I} & -\min \end{array}\right].$$

Because some constraints may have become redundant because of this, perform redundancy removal to obtain $[\Lambda_{\mathrm{B}}, \alpha_{\mathrm{B}}]$. An illustration of this procedure is given in Figure 6.

The straightforward procedures for Goal C are based on the similarities of the underlying problem with that of Goal A:

C1. Recall that the polytope of coherent lower previsions is the intersection of $|\mathcal{S}| = |\mathcal{K}| + 1$ polyhedra, one for each value of S. So apply a facet enumeration algorithm to the V-representation as given in Equation (13) for each S to obtain the corresponding H-representations $[A_S, b_S]$. An H-representation of the intersection polyhedron of polyhedra given as H-representations is the vertical concatenation of these matrices. (Intersection of polyhedra in V-representation, or mixed representations is not straightforward.) Perform redundancy removal on this concatenation H-representation to obtain $[\Lambda_{\mathrm{C}}, \alpha_{\mathrm{C}}]$. An illustration of this procedure is given in Figure 7.

C2. As can be seen from Equation (11), for each S we also know an H-representation for pairs $[\underline{P}; \mu_S]$ of which the $\underline{P}$-component belongs to the polyhedron corresponding to S already mentioned in Procedure C1:

$$\left[\begin{array}{cc|c} A_{S,\underline{P}} & A_{S,\mu_S} & b_0 \end{array}\right] \coloneqq \left[\begin{array}{cc|c} S & -SK^{\top} & 0 \\ & -\mathbb{I} & 0 \\ & 1^{\top} & 1 \\ & -1^{\top} & -1 \end{array}\right]. \tag{15}$$

Project this H-representation onto the $\underline{P}$-part to obtain the H-representation $[A_S, b_S]$ already encountered in Procedure C1, the remainder of which is to be followed here as well.

C3. Equation (11) also shows that we can actually create a single H-representation for pairs $[\underline{P}; \mu]$ of which the $\underline{P}$-components are coherent lower previsions:

$$\left[\begin{array}{cc|c} A_{\underline{P}} & A_{\mu} & b \end{array}\right] \coloneqq \left[\begin{array}{cc|c} A_{\mathbb{I},\underline{P}} & A_{\mathbb{I},\mu_{\mathbb{I}}} & b_0 \\ \vdots & \ddots & \vdots \\ A_{S_g,\underline{P}} & A_{S_g,\mu_{S_g}} & b_0 \\ \vdots & \ddots & \vdots \end{array}\right], \tag{16}$$

where $S_g \in \mathcal{S}$, with g in $\mathcal{K}$, has negative diagonal g-component. Projecting this H-representation onto the $\underline{P}$-part again gives us $[\Lambda_{\mathrm{C}}, \alpha_{\mathrm{C}}]$. Because of the block diagonal structure of the set of columns to be removed by projection, this procedure is essentially identical to Procedure C2 from the computational point of view.

Comparing the two main procedure types, enumeration-based (A1, B1, C1) and projection-based (A2, C2, C3), our numerical experiments showed that the enumeration-based ones were faster by at least an order of magnitude. It is not yet clear whether this is inherent or whether this is due to the fact that the enumeration implementation used (the double description method of Fukuda and Prodon [26]) is efficient, and the facet projection implementations used (Fourier–Motzkin and block elimination) are not.

4.2. A more involved type of procedure

All of the procedures in the previous section were based on Equations (5)–(6) and (11)–(12). In these expressions, $\underline{P}$ appears free, i.e., without being multiplied by a variable vector such as λ, this in contrast to the other expressions characterizing avoiding sure loss and coherence, Equations (3) and (9). This allowed us to consider $\underline{P}$ as variable as well, directly leading to the straightforward procedures.

In our earlier work [25], we created a procedure starting from the expressions with bound $\underline{P}$. This procedure is inefficient when compared to the best performing of the straightforward procedures presented above. However, it is possible to create bound-$\underline{P}$-based procedures that are relatively efficient; we present the ones we found here, as the techniques used might be useful in other contexts as well.

We first make an assumption, namely that all gambles are non-constant and non-negative with zero minimum, or NNZM. In Appendix 4.3 immediately following this section we show that for coherent lower previsions this assumption

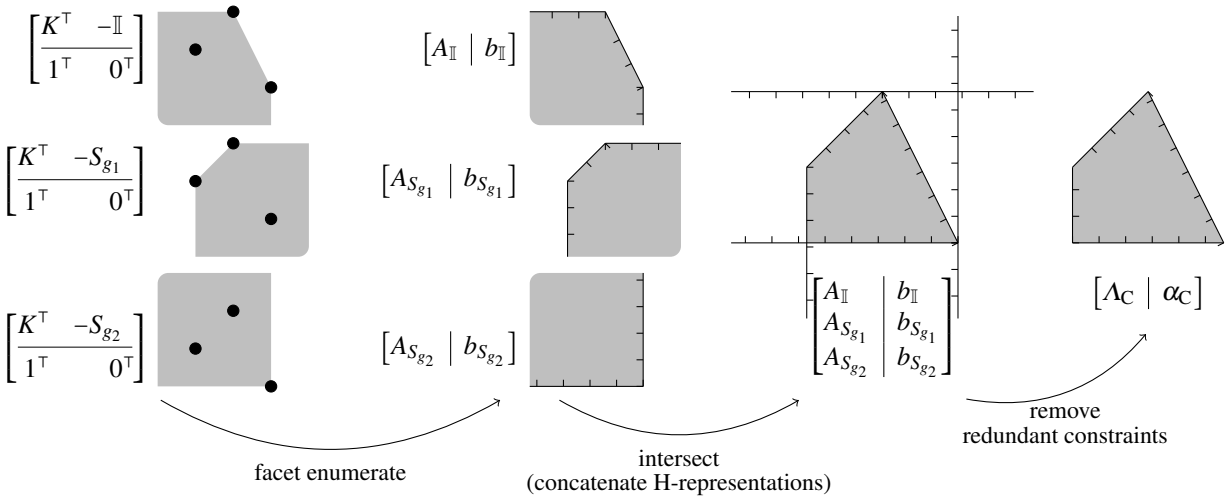

Figure 7: Illustration of Procedure C1 for the two-gamble example introduced in Figure 3, starting from the drawing of the polytope of coherent lower previsions first introduced in Figure 5.

is non-limiting and how to move between general gamble sets and NNZM gamble sets. The assumption is, however, limiting for lower previsions that only avoid sure loss. Note that $\underline{P} \geq \min$ becomes $\underline{P} \geq 0$ for an NNZM set of gambles $\mathcal{K}$; i.e., positivity constraints.

We do not develop procedures for Goal A here and move straight to Goal B, which because of the limiting nature of the NNZM assumption must be seen as preparation for the procedures for Goal C:

B2. We can rewrite Equation (3) as

$$\forall \gamma \in \mathbb{R} : \forall \lambda \geq 0 :\ \max(K\lambda) = \gamma \ \ \Rightarrow \ \ \underline{P}^\top \lambda \leq \gamma, \tag{17}$$

which, because $\mathcal{K}$ is NNZM, can be normalized to

$$\forall \lambda \geq 0 :\ \max(K\lambda) = 1 \ \ \Rightarrow \ \ \underline{P}^\top \lambda \leq 1. \tag{18}$$

Now, again because $\mathcal{K}$ is NNZM, $K\lambda$ is pointwise strictly increasing in λ. So we know that the feasible set $\{\lambda \geq 0 : K\lambda \leq 1\}$ is bounded and that apart from 0, all its vertices satisfy $\max(K\lambda) = 1$. So in our procedure, we first enumerate the vertices

$$[A \mid b] \coloneqq \left[\begin{array}{c|c} K & 1 \\ -\mathbb{I} & 0 \end{array}\right], \tag{19}$$

and then use this V-representation $[V; w]$ for the λ's to construct an H-representation $[V^\top, w^\top]$ for lower previsions; i.e., Equation (18) becomes $\underline{P}^\top V \leq w$, or $V^\top \underline{P} \leq w^\top$. As we are working towards Goal B, we also have to add positivity constraints $[-\mathbb{I}, 0]$; then after redundancy removal we obtain $[\Lambda_{\mathrm{B}}, \alpha_{\mathrm{B}}]$. An illustration of this procedure is given in Figure 8.

B3. Because we assume $\mathcal{K}$ is NNZM, $\underline{P} \geq 0$, so we know that all pointwise dominated vertices of the feasible set $\{\lambda \geq 0 : K\lambda \leq 1\}$ encountered in Procedure B2 result in redundant constraints (cf. the implicand in Equation (18)). So we can use the MOLP

$$\begin{aligned} &\text{maximize } \ \lambda, \\ &\text{subject to } \ K\lambda \leq 1 \ \text{ and } \ \lambda \geq 0, \end{aligned} \tag{20}$$

to select only the undominated vertices. Gather them as columns in a matrix $\hat{V}$ and construct the H-representation $[\hat{V}^\top, 1]$ to replace $[V^\top, w^\top]$ of Procedure B2. The advantage of this procedure over the preceding one is clarified on an example in Figure 8.

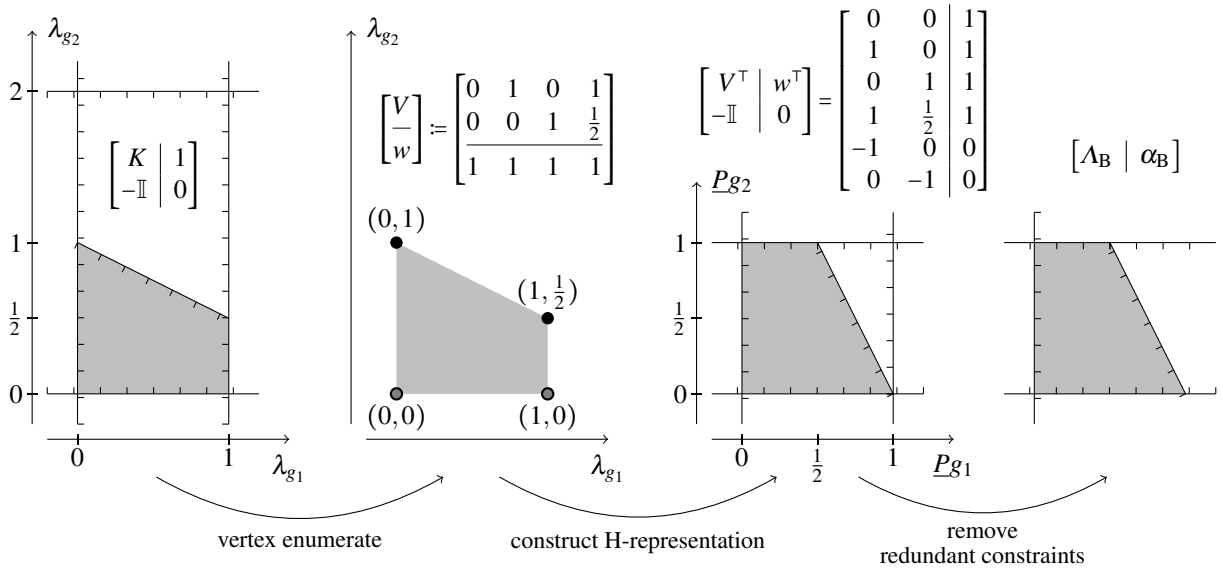

Figure 8: Illustration of Procedures B2, B3, and B4 for the two-gamble example introduced in Figure 3. Different from the preceding procedures, these procedures start in the space of λ-coefficients. In the second step, the transition is made to the space of lower previsions. The redundant constraints removed in the last step correspond to the first two rows of the preceding H-representation-matrix. These two redundant constraints correspond to the two dominated vertices of the feasible set in λ space, which are shown as gray filled dots (instead of being completely black). So the advantage of the MOLP-based Procedures B3 and B4 would be to avoid generating these vertices that result in redundant constraints.

B4. Because $K\lambda$ is pointwise strictly increasing in λ, we can replace the MOLP (20) by

$$\begin{aligned} &\text{maximize } K\lambda, \\ &\text{subject to } K\lambda \leq 1 \text{ and } \lambda \geq 0. \end{aligned} \tag{21}$$

The choice between this procedure and the preceding one depends on the size of the objective vector (λ and $K\lambda$, respectively): working with a small objective vector is computationally preferable.

We are now ready to present the procedures for Goal C, which strongly parallel those for Goal B:

C4. We can rewrite Equation (9) as

$$\forall S \in \mathcal{S} : \forall \lambda \geq 0 : \forall \gamma \in \mathbb{R} : \max(KS\lambda) = \gamma \;\Rightarrow\; \underline{P}^\top S\lambda \leq \gamma, \tag{22}$$

which, because $\mathcal{K}$ is NNZM and only a single column of KS is non-positive, but with zero maximum, can be normalized and rewritten as

$$\forall S \in \mathcal{S} : \forall \kappa \in \mathbb{R}^{\mathcal{K}} : S\kappa \geq 0 \Rightarrow \begin{cases} \max(K\kappa) = 1 \;\Rightarrow\; \underline{P}^\top \kappa \leq 1, \\ \max(K\kappa) = 0 \;\Rightarrow\; \underline{P}^\top \kappa \leq 0. \end{cases} \tag{23}$$

Now, again because $\mathcal{K}$ is NNZM, $K\kappa$ is pointwise strictly increasing in κ. So we know that the set $\{S\kappa \geq 0 : K\kappa \leq 1\}$ is bounded and that apart from 0, all its vertices satisfy $\max(K\kappa) = 1$. We also know that the set $\{0 \leq S\kappa \leq 1 : K\kappa \leq 0\}$ is bounded and that all its vertices satisfy $\max(K\kappa) = 0$. So the procedure consists in, for every S in $\mathcal{S}$, vertex enumerating

$$\left[A_{S,0} \mid b_{S,0}\right] \coloneqq \left[\begin{array}{c|c} K & 0 \\ -S & 0 \\ S & 1 \end{array}\right], \qquad \left[A_{S,1} \mid b_{S,1}\right] \coloneqq \left[\begin{array}{c|c} K & 1 \\ -S & 0 \end{array}\right]; \tag{24}$$

then use the resulting V-representations $[V_{S,1}; w_{S,1}]$ and $[V_{S,0}; w_{S,0}]$ to construct the H-representations $[V^\top_{S,1}, w^\top_{S,1}]$ and $[V^\top_{S,0}, 0]$. Vertically concatenate these H-representations for every S to obtain an H-representation for the set of coherent lower previsions on $\mathcal{K}$ and apply redundancy removal to obtain $[\Lambda_{\mathrm{C}}, \alpha_{\mathrm{C}}]$.

Entirely analogously to what was done in Procedures B3 and B4, we can use MOLPs to generate undominated vertex versions of $[V_{S,\gamma}; w_{S,\gamma}]$ for all S in $\mathcal{S}$ and γ in $\{0,1\}$:
C5. The κ-variant:

$$\begin{aligned} &\text{maximize} && \kappa, \\ &\text{subject to} && K\kappa \leq \gamma,\ S\kappa \geq 0 \text{ and, if } \gamma = 0,\ S\kappa \leq 1. \end{aligned} \tag{25}$$

C6. The $K\kappa$-variant:

$$\begin{aligned} &\text{maximize} && K\kappa, \\ &\text{subject to} && K\kappa \leq \gamma,\ S\kappa \geq 0 \text{ and, if } \gamma = 0,\ S\kappa \leq 1. \end{aligned} \tag{26}$$

In principle, the MOLP-based procedures (B3, B4, C5, and C6) should be more efficient than the vertex enumeration ones (B2, C4), as for both the same polytope needs to be mapped, but for the MOLPs only in part, which also results in less redundant constraints to be removed later on. In our numerical experiments, the vertex enumeration variant turned out to be quite efficient: the number of redundant constraints it produces is about the same as the number of non-redundant ones; for our earlier procedure, this quickly grew beyond a difference of an order of magnitude. However, the results for Procedures B3 and C5 were not as good: the M3-solver at our disposal [27] could not deal in reasonable time with sets of gambles that the enumeration-based procedures digested almost instantly (its author explained that it was not designed for large objective vectors). Procedures B4 and C6 could not be tested due to an apparent lack of publicly available M4-solvers.

4.3. Appendix: the NNZM assumption & coherence

Given a general set of gambles $\mathcal{K}$, let $\bar{\mathcal{K}}$ be the subset of constant gambles and $\check{\mathcal{K}}$ the subset of non-constant gambles. Let $\bar{b}$ be the vector with the values of the constant gambles and $\hat{\mathcal{K}}$ an NNZM set of gambles associated with $\check{\mathcal{K}}$. The restrictions of a lower prevision $\underline{P}$ on $\bar{\mathcal{K}} \cup \check{\mathcal{K}} \cup \hat{\mathcal{K}}$ to these sets are $\bar{\underline{P}}$, $\check{\underline{P}}$, and $\hat{\underline{P}}$. (Properties of coherent lower previsions used here can be found in Walley's book [1, §2.6.1(b),(c)]).

If $\underline{P}$ is coherent, we know that $\underline{P}\beta = \beta$ for any constant gamble β and so the constraints are $\bar{\underline{P}} = \bar{b}$. For any other gamble f in $\mathcal{K}$ we have the linking constraint $\check{\underline{P}} f - \hat{\underline{P}}(f - \min f) = \min f$. Fix $\hat{\mathcal{K}} \coloneqq \{f - \min f : f \in \check{\mathcal{K}}\}$; this set is NNZM. Let $\hat{A}\hat{\underline{P}} \leq \hat{b}$ be the constraints for the polytope of coherent lower previsions $\hat{\underline{P}}$ on $\hat{\mathcal{K}}$, then, using the linking constraints, the corresponding constraints for $\check{\underline{P}}$ on $\check{\mathcal{K}}$ are $\hat{A}\check{\underline{P}} \leq \hat{b} + \hat{A}\min$. So the full H-representation of the set of coherent lower previsions $[\bar{\underline{P}}; \check{\underline{P}}]$ on $\mathcal{K}$ is

$$\left[A_{\mathcal{K}} \mid b_{\mathcal{K}}\right] \coloneqq \left[\begin{array}{cc|c} \mathbb{I} & & \bar{b} \\ -\mathbb{I} & & -\bar{b} \\ & \hat{A} & \hat{b} + \hat{A}\min \end{array}\right]. \tag{27}$$

4.4. Quantitative results of numerical experiments

Above, we have already mentioned some qualitative evaluations and comparisons of the different procedures. Here we present more quantitative results. Our data points are obtained from randomly generated NNZM gamble sets with values taken from $\{0, \ldots, 9\}$. The absolute magnitude of the obtained running times give an indication of what problems can be dealt with in practice at the time of writing. (Our CPU-bound numerical (floating point) experiments were run on an Intel i7-2620M processor.) The relative running times for different parameter values provide indications—but not certainty—about the computational complexity of the procedures. (The Python scripts we developed are publicly available [28].)

Our experiments showed that the *sparsity* σ, i.e., the fraction of zero components in the gamble matrix K, has an important influence on the running times of our procedures. The graph in Figure 9 indicates that the running time of Procedure C1

i. decreases exponentially as a function of the sparsity,
ii. increases approximately linearly as a function of $|\Omega|$, because the curves of doubling possibility space cardinality $|\Omega|$ are approximately equidistant.

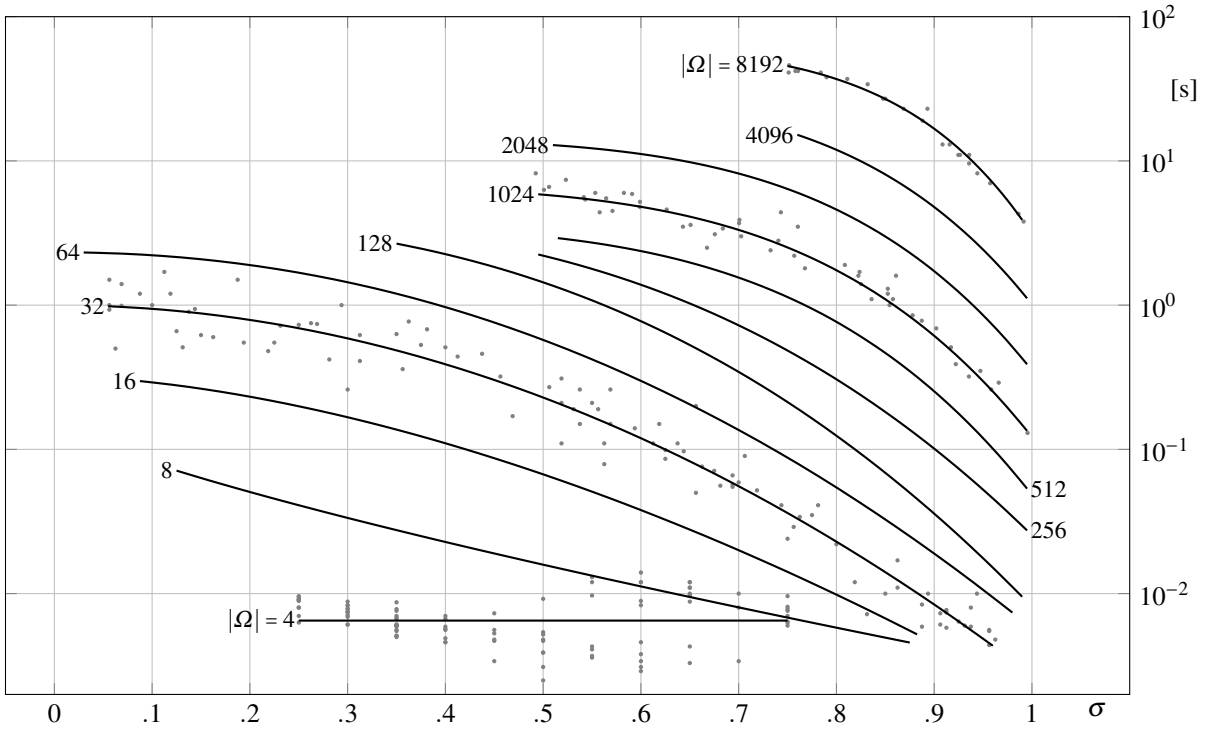

Figure 9: Running times for Procedure C1 for $|\mathcal{K}| = 5$ and different cardinalities of Ω. The curves are Levenberg–Marquardt least-squares fits of $a \cdot \exp(-b \cdot \sigma^c)$ to the data points, with real fit parameters $a, b, c > 0$. To give an idea of the variance, we have also plotted the data points for $|\Omega|$ in $\{4, 32, 1024, 8192\}$ as gray dots. Approximate number of data points per cardinality: 100 for $|\Omega| \in \{4, 8, 16, 32, 64\}$, 50 for $|\Omega| \in \{128, 256, 512, 1024\}$, and 25 for $|\Omega| \in \{2048, 4096, 8192\}$.

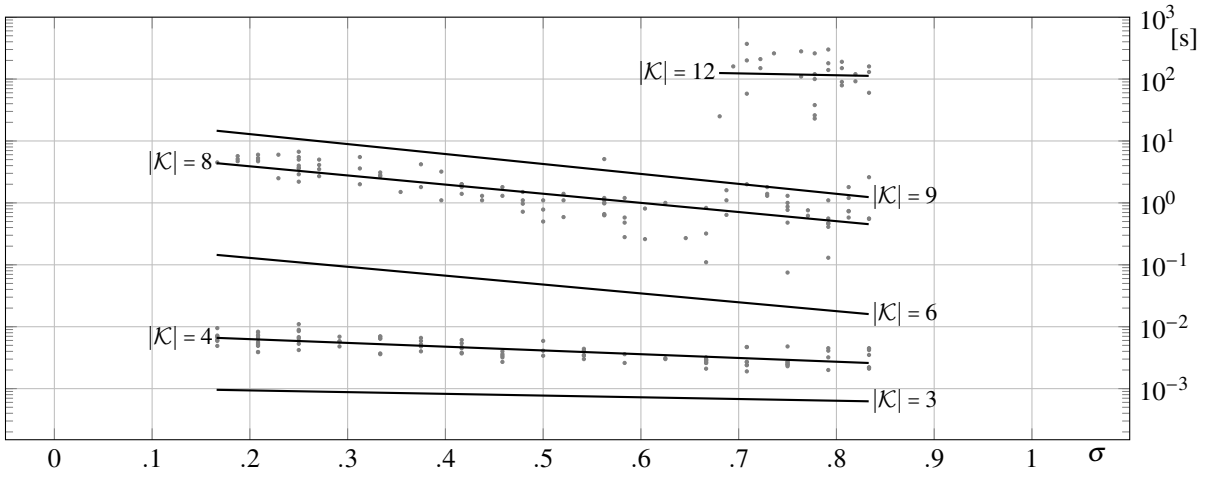

Figure 10: Running times for Procedure C1 for $|\Omega| = 6$ and different cardinalities of $\mathcal{K}$. The curves are (logarithmic) least-squares fits of $a \cdot \exp(-b \cdot \sigma)$ to the data points, with real fit parameters $a, b > 0$. To give an idea of the variance, we have also plotted the data points for $|\mathcal{K}|$ in $\{4, 8, 12\}$ as gray dots. Approximate number of date points per cardinality: 100 for $|\mathcal{K}| \in \{3, 4, 6, 8\}$, 75 for $|\mathcal{K}| = 9$, and 25 for $|\mathcal{K}| = 12$.

The same gamble sets were also processed using Procedure C4; the running times were typically 1.5 times, but sometimes 4 times longer. The other procedures were orders of magnitude too slow for reliable testing.

The graph in Figure 10 indicates that the running time of Procedure C1 increases (at least) exponentially as a function of $|\mathcal{K}|$, because of the approximate equidistance of the lines for $|\mathcal{K}|$ in $\{3, 6, 9\}$ and for $|\mathcal{K}|$ in $\{4, 8, 12\}$, respectively.

5. Correcting incoherent lower previsions

Now that we have procedures for obtaining minimal linear constraint characterizations for lower previsions that avoid sure loss or are coherent, we are ready to look at what lies beyond: sure loss and other forms of incoherence.

Automatic methods for learning lower previsions from data ideally produce coherent lower prevision, but some may not—possibly for good reasons. Also, when eliciting lower previsions from experts—but not in imprecise probability theory—, it is not reasonable to expect the result to be coherent or perhaps even avoid sure loss. For incoherent, but sure loss avoiding lower previsions, we can apply natural extension to perform a pointwise upward correction that makes explicit all implicit commitments. This is appropriate when the user of the automatic method or the elicitee provide informed consent. Otherwise a conservative, downward correction may be more acceptable.

Downward changes of a lower prevision imply a reduction in both explicit and implicit commitments. When it is not possible to decide on the changes with input from the user or the elicitee, automatic downward correction methods are an option, after informed consent. We here propose one such automatic downward correction method.

5.1. Forms of Incoherence

Let us briefly give a categorization of the possible forms of incoherence.

We recalled at the end of Section 3 that coherent lower previsions $\underline{P}$ are bounded, i.e., that $\min \leq \underline{P} \leq \max$. Our first category of incoherent previsions are those that are out of bounds; it is illustrated in Figure 11. The violation of these constraints can be very easily checked.

For lower previsions within bounds, one can consider incurring sure loss to be the most severe form of incoherence; this class is also illustrated in Figure 11. To check whether a given lower prevision incurs sure loss or not, a linear program such as the one given in Equation (4) or (5) needs to be solved.

Lower previsions that avoid sure loss can still be incoherent if they incur sure S-loss for some S in $\mathcal{S}$; also this class is illustrated in Figure 11. So to check whether a given lower prevision is incoherent or not, we need to solve $|\mathcal{S}|$ linear programs such as the ones given in Equation (10) or (11). It is possible that a lower prevision incurs sure S-loss for all S in $\mathcal{S}$; e.g., max in our three-dimensional running example, as can be seen in Figure 5.

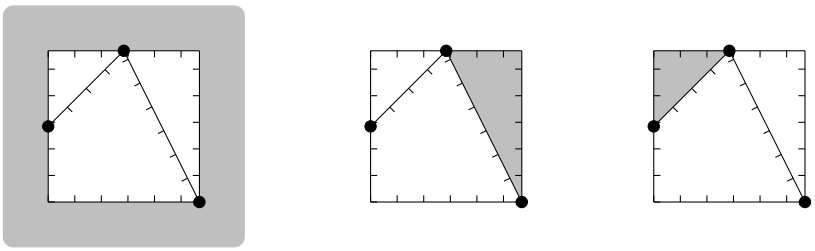

Figure 11: An illustration of different forms of incoherence for the example introduced in Figure 3. In the figure on the left, the magnitude-wise smallest part of the unbounded region of out-of-bounds lower previsions is shaded gray. In the middle figure, the lower previsions within bounds that incur sure loss are shaded gray. In the figure on the right, the incoherent lower previsions within bounds that avoid sure loss are shaded gray.

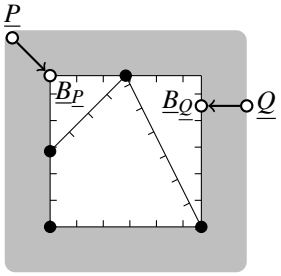

Figure 12: Illustration of the proposed method to bring out of bounds lower previsions within bounds. It shows, for the example introduced in Figure 3, how two out of bounds lower previsions, $\underline{P}$ and $\underline{Q}$, are corrected. The lower previsions of interest are shown as white-filled dots.

5.2. Bringing lower previsions within bounds

Correcting a lower prevision $\underline{P}$ that is out of bounds to one that is within bounds is trivial: We replace it by the pointwise closest such lower prevision $\underline{B}_{\underline{P}}$, so for every gamble f in $\mathcal{K}$ we have

$$\underline{B}_{\underline{P}} f := \begin{cases} \min f & \underline{P} f \leq \min f, \\ \max f & \underline{P} f \geq \max f, \\ \underline{P} f & \text{otherwise.} \end{cases} \tag{28}$$

This correction method may produce both downward and upward pointwise changes. It is illustrated in Figure 12.

From now on we assume that all lower previsions are within bounds.

5.3. Maximal dominated coherent lower previsions

Our proposal for the downward correction of an incoherent lower prevision $\underline{P}$ is the lower envelope of the maximal coherent lower previsions dominated by $\underline{P}$. In other words, it is the nadir point $\underline{D}_{\underline{P}}$ of the MOLP (cf. Section 2.2):

$$\begin{aligned} &\text{maximize} && \underline{Q}, \\ &\text{subject to} && \Lambda_{\mathrm{C}} \underline{Q} \leq \alpha_{\mathrm{C}} \quad \text{and} \quad \underline{Q} \leq \underline{P}. \end{aligned} \tag{29}$$

This proposal is essentially the same as the specific so-called *prudential correction* $\overline{P}_H$ mentioned by Pelessoni and Vicig [29, §3.4]. They generalize Weichselberger's interval-probability concept *F-Hülle* [30, 342ff. and 375ff.]—translated as *F-cover* in a paper of his [31]. However, they only aim to apply this correction when sure loss is avoided; we make no such restriction. The correction method is illustrated in Figure 13.

The lower prevision $\underline{D}_{\underline{P}}$ satisfies the necessary requirements:

i. It is a downward correction as a lower envelope of lower previsions dominated by $\underline{P}$.
ii. It is coherent by the Lower Envelope Theorem.

Furthermore, as a nadir point it has a number of further desirable properties:

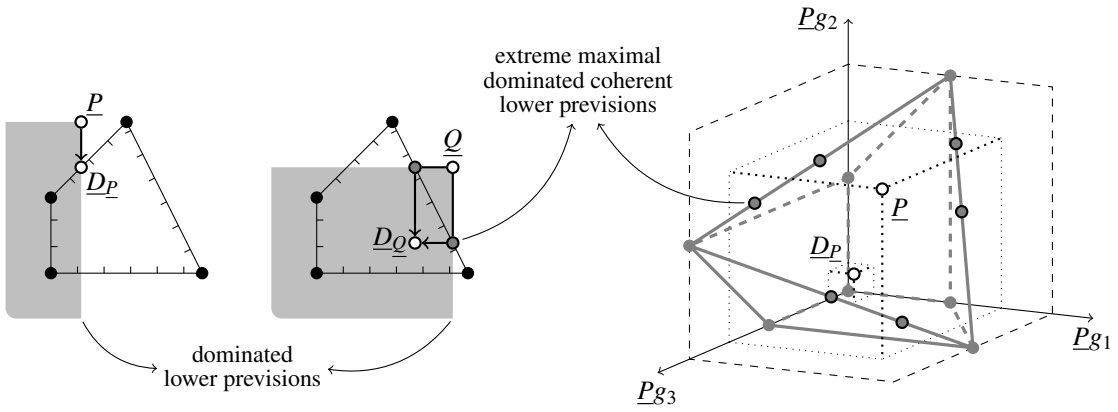

Figure 13: An illustration of the proposed downward correction method for the examples introduced in Figure 3. On the left, for the two-dimensional example, two incoherent lower previsions are corrected, $\underline{P}$ and $\underline{Q}$. On the right, for the three-dimensional example, one incoherent lower prevision is corrected, $\underline{P}$. Again, the lower previsions of interest are shown as white-filled dots. Extreme maximal dominated coherent lower previsions are shown as gray-filled dots. For the two-dimensional case, the sets of dominated lower previsions are shaded in gray. The three-dimensional case shows that the extreme maximal coherent lower previsions dominated by the given incoherent lower prevision cannot always be reached by reducing single components, i.e., by projection along the canonical axes onto the polytope of coherent lower previsions.

iii. The correction it embodies is neutral in the sense that no trade-off between corrections for the different components of $\underline{P}$ is made; this makes it especially suited for unguided corrections.
iv. It is the maximal such neutral correction—the vacuous lower prevision min is another—and therefore preserves as much of the commitments expressed by $\underline{P}$ as possible.
v. The set of coherent lower previsions dominated by an incoherent lower prevision $\underline{P}$ is non-decreasing with pointwise increasing $\underline{P}$. So the more incoherent a lower prevision, the more imprecise its correction.

It is actually not necessary to calculate $[\Lambda_C, \alpha_C]$ in order to find $\underline{D}_{\underline{P}}$, because we have a full constraint based characterization of coherence with the H-representation (16). So an alternative to the MOLP (29) is the following MOLP:

$$\begin{aligned} &\text{maximize } \underline{Q}, \\ &\text{subject to } A_{\underline{Q}}\underline{Q} + A_\mu \mu \le b \text{ and } \underline{Q} \le \underline{P}, \end{aligned} \tag{30}$$

where we use the notation of Equation (16). (Weichselberger [30, 468ff] also proposes an as of yet untested algorithm that is essentially based on a representation such as the one given by Equation (16).) This problem has $(|\mathcal{K}|+1)\cdot|\Omega|$ more variables than the MOLP (29), which has $|\mathcal{K}|$ variables. It has $(|\mathcal{K}|+1)\cdot(|\mathcal{K}|+|\Omega|+2)$ constraints, whereas the MOLP (29) typically has of the order of $3\cdot|\mathcal{K}|$ constraints. This results in a greater average running time for the nadir computation using the alternative MOLP, even if we take the setup time—calculating $[\Lambda_C, \alpha_C]$ (cf. Section 4.4) versus generating $[A_{\underline{Q}}, A_\mu, b]$ (about 10^{-3}s at present)—into account. This can be seen in the graphical summary of the results of our numerical experiments in Figure 14. (The Octave/Matlab scripts we developed are publicly available [32].)

With the M3-solver we used [27], average computation time seems to increase exponentially as a function of $|\mathcal{K}|$. Surprisingly, the number of extreme maximal dominated coherent lower previsions is not a major factor. This is illustrated by the number of these extreme points found for the minimum and maximum computation times—put in italics near the respective box plot whiskers—and the maximum number of extreme points in the sample—listed in italics at the top edge of the plot axis.

The M3-solver does compute all these extreme points, so we suspect that it is highly inefficient for the task at hand. Therefore we believe substantial efficiency gains can be achieved by switching to an M4-solver, which we expect to be *output sensitive*, i.e., to depend on the number of extreme points. Nadir point calculation algorithms that do not need to calculate all these extreme points such as the one by [13] should provide a further increase in efficiency. Because elicited lower previsions can be expected to generally be closer to coherent than our randomly generated ones, we also expect them to generally dominate less extreme points and thus, because of output sensitivity, be faster to correct. We already observed this phenomenon for randomly generated sure loss avoiding lower previsions.

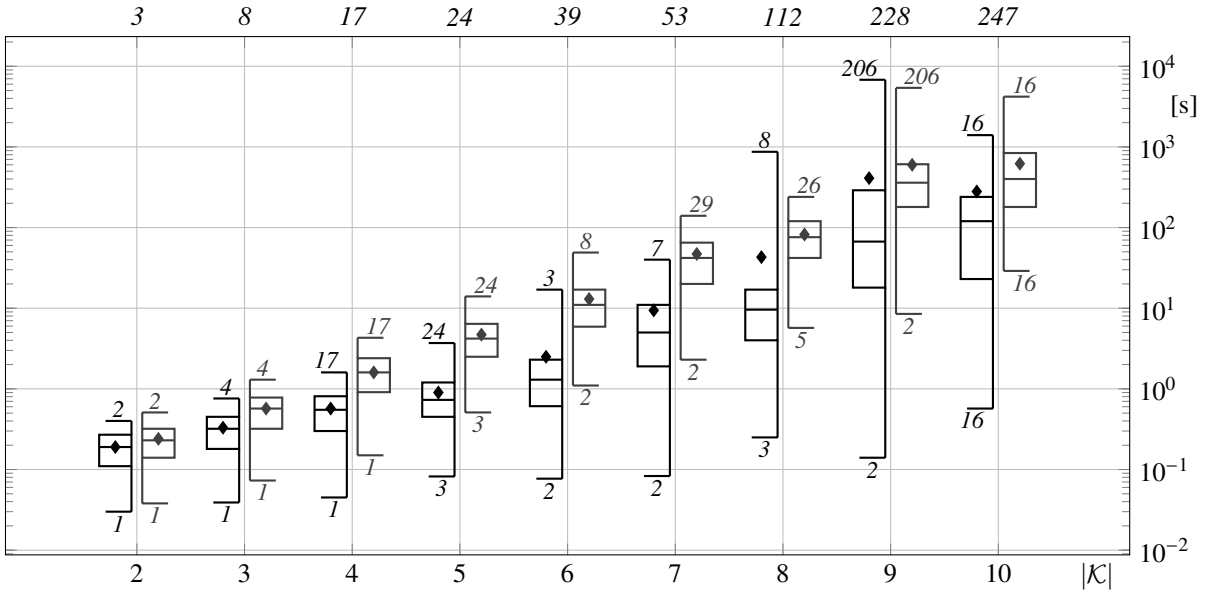

Figure 14: Graphical summary of the numerical coherence correction experiments. In the experiments, we used $|\Omega| = 5$ and $\sigma \approx 1/2$ and looked at the running time to calculate $\underline{D_P}$ for $|\mathcal{K}|$ from 2 to 10. We used about 100 samples for each value of $|\mathcal{K}|$: we generated about 10 NNZM gamble sets $\mathcal{K}$, for each of which we calculated the corresponding $[\Lambda_C, \alpha_C]$—using Procedure C1—and generated the corresponding $[A_{\underline{Q}}, A_\mu, b]$; finally, for each of these $\mathcal{K}$ we generated about 10 incoherent lower previsions within bounds to correct. Black left-leaning box plots are used for the results obtained with the MOLP (29); dark-gray right-leaning ones for those obtained with the MOLP (30). Each of these sample sets is summarized using a box plot indicating minimum, lower quartile, median, upper quartile, and maximum; its arithmetic mean is indicated with a lozenge.

5.4. Least dominating coherent lower prevision

For completeness's sake, let us also have a look at upward correction using the MOLP approach. Given an incoherent lower prevision $\underline{P}$, we consider the set of minimal pointwise dominating coherent lower previsions; this is the solution to the following MOLP:

$$\begin{aligned} &\text{minimize } \underline{E}_{\underline{P}}, \\ &\text{subject to } \Lambda_C \underline{E}_{\underline{P}} \leq \alpha_C \text{ and } \underline{E}_{\underline{P}} \geq \underline{P}. \end{aligned} \tag{31}$$

Because of the Lower Envelope Theorem, there is only one such $\underline{E}_{\underline{P}}$, so we may replace this vector objective by the scalar objective $\sum_{g \in \mathcal{K}} \underline{E}_{\underline{P}} g$, reducing the problem to a plain LP. This coherent lower prevision $\underline{E}_{\underline{P}}$ is the one least dominating $\underline{P}$, to wit, its natural extension [1, §3.1]. This plain LP method for obtaining it is illustrated in Figure 15.

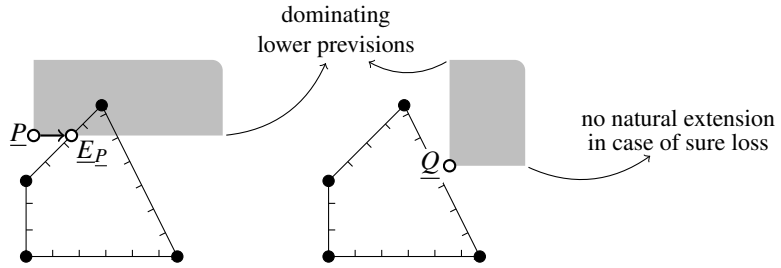

Figure 15: An illustration of natural extension for the two-dimensional example introduced in Figure 3. Two incoherent lower previsions are considered. One, $\underline{P}$, avoids sure loss and can be upwardly corrected. The other, $\underline{Q}$, does not avoid sure loss and can therefore not be upwardly corrected. As before, the lower previsions of interest are shown as white-filled dots.

Again, we can use the H-representation (16) to formulate an alternative to the MOLP (31):

$$\begin{aligned} &\text{minimize } \underline{E}_{\underline{P}}, \\ &\text{subject to } A_{\underline{E}_{\underline{P}}}\underline{E}_{\underline{P}} + A_{\mu}\mu \leq b \text{ and } \underline{E}_{\underline{P}} \geq \underline{P}. \end{aligned} \tag{32}$$

Thanks to the block structure of the constraint matrix, it is straightforward to deduce some well-known facts:

i. It is necessary that $\underline{P}$ avoids sure loss for a solution $\underline{E}_{\underline{P}}$ to exist.
ii. For each gamble g in $\mathcal{K}$, we can calculate the corresponding natural extension component $\underline{E}_{\underline{P}}g$ separately as $\max\{g^{\top}\mu : \underline{P} \leq K^{\top}\mu \wedge \mu \geq 0 \wedge 1^{\top}\mu = 1\}$.

These facts raise the currently still open question of whether there exist specific classes of incoherent lower previsions $\underline{P}$ for which the calculation of $\underline{D}_{\underline{P}}$ can be simplified, e.g., to separate calculations for each component.

6. Conclusions

We hope that you are now convinced of the fact that the availability of a finite, minimal linear constraints characterization of coherence opens doors for many new numerical applications dealing with the set of coherent lower previsions. In our application, downward correction of incoherent lower previsions, we saw that it proved useful to keep the running time of the inherently computationally complex implementation of our proposed method a bit in check. We determined that currently, sets of up to 5 gambles can be dealt with sufficiently fast even for interactive applications. In a domain where complex systems are often decomposed into smaller ones linked in some network structure, this is not overly restrictive.

We also hope that this paper has kindled your interest in the application of multi-objective linear programming to imprecise probability problems. We believe that beyond the two applications of them presented in this paper, there are bound to be more in our research field because of the common underlying assumption that incomparability should be modeled, not avoided.

There are some unfinished strands in this paper:

i. Testing an efficient projection implementation [9].
ii. Finding and testing a MOLP simplex solver (cf. M4) and a nadir computation algorithm (cf. M2).
iii. Theoretically investigating whether $\underline{D}_{\underline{P}}$ can be calculated more efficiently if $\underline{P}$ satisfies some additional conditions beyond being within bounds.

We hope these are picked up by us, or others, in the future.

Acknowledgments

Thanks to the reviewers of the different versions of this paper for pointing out very relevant literature and suggestions for improvement. Thanks to J. De Bock for providing me with extracts from Weichselberger's tome [30] and Th. Augustin for helping me interpret them correctly. Thanks to G. de Cooman and N. Huntley for sharing their thoughts about the downward correction idea. Thanks to A. P. Pedersen for pointers to potentially relevant belief revision texts, which, I am sorry to say, I could not yet follow up on.

References

[1] P. Walley, Statistical reasoning with imprecise probabilities, volume 42 of *Monographs on Statistics and Applied Probability*, Chapman & Hall, London, 1991.
[2] E. Miranda, A survey of the theory of coherent lower previsions, International Journal of Approximate Reasoning 48 (2008) 628–658.
[3] B. Grünbaum, Convex polytopes, Interscience Publishers, London, 1967.
[4] G. M. Ziegler, Lectures on polytopes, Springer, 1995.
[5] K. Fukuda, Frequently asked questions in polyhedral computation, 2004. URL: http://www.ifor.math.ethz.ch/~fukuda/polyfaq.
[6] K. L. Clarkson, More output-sensitive geometric algorithms, in: Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 1994, p. 695–702. doi:10.1109/SFCS.1994.365723.
[7] D. Avis, K. Fukuda, A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra, Discrete & Computational Geometry 8 (1992) 295–313.

[8] C. N. Jones, E. C. Kerrigan, J. M. Maciejowski, Equality Set Projection: A new algorithm for the projection of polytopes in halfspace representation, Technical Report CUED/F-INFENG/TR.463, Department of Engineering, University of Cambridge, 2004. URL: http://www-control.eng.cam.ac.uk/~cnj22/docs/resp_mar_04_15.pdf.
[9] M. Kvasnica, P. Grieder, M. Baotić, Multi-parametric toolbox (mpt), version 2.6.3, 2006. URL: http://control.ee.ethz.ch/~mpt.
[10] M. Ehrgott, Multicriteria optimization, 2 ed., Springer, 2005.
[11] D. Bertsimas, J. N. Tsitsiklis, Introduction to linear optimization, Athena Scientific, 1997. URL: http://athenasc.com/linoptbook.html.
[12] M. Ehrgott, D. Tenfelde-Podehl, Computation of ideal and nadir values and implications for their use in MCDM methods, European Journal of Operational Research 151 (2003) 119–139.
[13] M. J. Alves, J. P. Costa, An exact method for computing the nadir values in multiple objective linear programming, European Journal of Operational Research 198 (2009) 637–646.
[14] H. P. Benson, An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem, Journal of Global Optimization 13 (1998) 1–24.
[15] M. Ehrgott, A. Löhne, L. Shao, A dual variant of Benson's "outer approximation algorithm" for multiple objective linear programming, Journal of Global Optimization 52 (2012) 757–778.
[16] J. P. Evans, R. E. Steuer, A revised simplex method for linear multiple objective programs, Mathematical Programming 5 (1973) 54–72.
[17] L. W. Strijbosch, A. G. van Doorne, W. J. Selen, A simplified MOLP algorithm: The MOLP-S procedure, Computers & Operations Research 18 (1991) 709–716.
[18] P. L. Yu, M. Zeleny, The set of all nondominated solutions in linear cases and a multicriteria simplex method, Journal of Mathematical Analysis and Applications 49 (1975) 430–468.
[19] H. Isermann, The enumeration of the set of all efficient solutions for a linear multiple objective program, Operational Research Quarterly 28 (1977) 711–725.
[20] P. Walley, R. Pelessoni, P. Vicig, Direct algorithms for checking consistency and making inferences from conditional probability assessments, Journal of Statistical Planning and Inference 126 (2004) 119–151.
[21] P. Vicig, An algorithm for imprecise conditional probability assessments in expert systems, in: IPMU '96: Proceedings of the Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, 1996, p. 61–66.
[22] V. Biazzo, A. Gilio, A generalization of the fundamental theorem of de Finetti for imprecise conditional probability assessments, International Journal of Approximate Reasoning 24 (2000) 251–272.
[23] E. Quaeghebeur, G. De Cooman, Extreme lower probabilities, Fuzzy Sets and Systems 159 (2008) 2163–2175.
[24] E. Quaeghebeur, Learning from samples using coherent lower previsions, Ph.D. thesis, Ghent University, 2009. URL: http://hdl.handle.net/1854/LU-495650. arXiv:1854/LU-495650.
[25] E. Quaeghebeur, Characterizing the set of coherent lower previsions with a finite number of constraints or vertices, in: P. Spirtes, P. Grünwald (Eds.), UAI-10: Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, AUAI Press, 2010, p. 466–473. URL: http://hdl.handle.net/1854/LU-984156. arXiv:1854/LU-984156.
[26] K. Fukuda, A. Prodon, Double description method revisited, in: M. Deza, R. Euler, I. Manoussakis (Eds.), Combinatorics and Computer Science, volume 1120 of *Lecture Notes in Computer Science*, Springer-Verlag, 1996, p. 91–111. URL: http://www.ifor.math.ethz.ch/~fukuda/cdd_home. doi:10.1007/3-540-61576-8_77.
[27] A. Löhne, bensolve, version 1.2, 2012. URL: http://ito.mathematik.uni-halle.de/~loehne/index_en_dl.php.
[28] E. Quaeghebeur, pycohconstraints: Python code for generating coherence constraints for lower previsions, 2013. URL: http://github.com/equaeghe/pycohconstraints.
[29] R. Pelessoni, P. Vicig, Imprecise previsions for risk measurement, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 11 (2003) 393–412.
[30] K. Weichselberger, Elementare Grundbegriffe einer allgemeineren Wahrscheinlichkeitsrechnung I: Intervallwahrscheinlichkeit als Umfassendes Konzept, Physica-Verlag, Heidelberg, 2001.
[31] K. Weichselberger, The theory of interval-probability as a unifying concept for uncertainty, International Journal of Approximate Reasoning 24 (2000) 149–170.
[32] E. Quaeghebeur, mcohconstraints: Matlab/Octave functions for generating coherence and avoiding sure loss constraints for lower previsions, 2013. URL: http://github.com/equaeghe/mcohconstraints.