Conservative decision-making with sets of probabilities: How to infer new choices from previous ones

Arne Decadt, Alexander Erreygers and Jasper De Bock

Foundations Lab for imprecise probabilities, Technologiepark-Zwijnaarde 125, Zwijnaarde, 9052, Flanders, Belgium

Abstract

We study a generalized version of maximizing expected utility, called E-admissibility, to make decisions when the decision-maker's uncertainty is described by a set of probability mass functions. In particular, instead of specifying this set directly, we assume that we only have partial information about the decision-maker's preferences or choices, in the form of which options she rejects from some finite sets of options. We describe both the decision-making process and the available information using choice functions, and we provide an algorithm, based on linear programming, to compute the most conservative extension of a given choice assessment to a choice function that makes decisions based on E-admissibility. Next, we relate this E-admissible extension to the so-called coherent extension and show how the same techniques that are used to simplify the computation of this coherent extension can also be used to simplify that of the E-admissible one. In our experiments, we demonstrate that decision-making with the E-admissible extension is faster and more informative than with the coherent one, but also observe that the required computations are challenging once the parameters of the problem scale.

Keywords: E-admissible extension, sets of probabilities, E-admissibility, choice function, decision-making, linear programming

1. Introduction

A decision-maker's uncertainty is typically modelled by a probability measure, and it is often argued that her rational decisions should maximise expected utility with respect to this probability measure. However, she may not always have sufficient knowledge to come up with a unique and completely specified probability measure. It is then often assumed, as a work-around, that there is some set of probability measures—also called a credal set—that describes her uncertainty. In this setting, the term *E-admissibility* was coined by Levi [1, 2] for a decision criterion suggested by Good [3]. It is but one of several generalisations of the standard criterion of maximising expected utility, where the decision-maker now only keeps those options that maximise expected utility with respect to at least one of the probability measures in the credal set, and rejects all the others.[1]

We here study decision-making with this criterion when the credal set is not specified directly, but is instead derived from (partial information about) the decision-maker's past choices. We start in Section 2 by introducing the mathematical framework of choice functions as a tool to model decision-making, and then go on in Section 3 by introducing the standard criterion of maximising expected utility as well as the E-admissibility criterion. Next, in Section 4, we explain how information about earlier decisions—an assessment—can be seen as a partial specification of a choice function and study if it is possible to extend

[1] Levi's [1] original definition considered credal sets that are convex, whereas we do not require this. In fact, one of the strengths of our approach is that an assessment can lead to non-convex credal sets; see the example in Example 6.2 further on.

this partial specification to a choice function that chooses based on E-admissibility from any given option set. In case this is possible, we look at what the most conservative such extension is for a given assessment, and call this the E-admissible (natural) extension. In Section 5 we then characterise the set of probability mass functions that agree with an assessment. Readers that are familiar with these concepts can safely skim through these two sections, focussing mainly on the notation that we introduce. Next, in Section 6, we show how we can use this to evaluate the E-admissible extension of an assessment, and provide an algorithm that can compute this E-admissible extension for arbitrary finite assessments. Section 7 is concerned with an alternative, more conservative method for extending choice assessments, called the coherent (natural) extension[2], which we briefly introduce and compare to the E-admissible extension. Section 8 exploits the connection between both extensions to enable us to apply existing simplification methods for the coherent extension to the E-admissible extension as well. Finally, in our experiments in Section 9, we study when it is advantageous to use the simplifications of Section 8, and we compare the E-admissible extension to the coherent extension for both the evaluation time and number of chosen options. In the appendix, we have included a table of notation so that the reader can quickly look up notation that they might have missed or forgotten.

Choice functions are also often used in economic contexts and revealed preference theory [6]. Most of this field however focuses on getting a single preference relation from a choice function, while we work with a set of preference relations. In particular, our model and choice functions do not necessarily satisfy the weak axiom of revealed preference. Related to this, our assessments allow for more general statements about than what is usually used in revealed preference theory, although our options themselves are more restrictive. Some more recent work in this area, such as [7] investigates set-rationalisable choice functions of which our choice functions are a special case, but from a more theoretical perspective.

We are not the first to develop algorithms for decision-making with E-admissibility. Utkin and Augustin [8] and Kikuti et al. [9] have gone before us, but they focus on a setting where the credal set is specified directly by means of a conjunction of inequality constraints, while we specify them indirectly using partial information about choice functions [10, 11, 12]. Decadt et al. [4] have also studied such more general assessments, but for the coherent (natural) extension. We compare our approach to that of Decadt et al. [4] in Sections 7 and 9. We revisit the connection with Utkin and Augustin [8] and Kikuti et al. [9] in our conclusions.

This paper extends an earlier conference contribution of the authors [13]. The main novelties with respect to that contribution are the discussion of the connection with the coherent (natural) extension in Section 7, the simplification that can be achieved by using this connection in Section 8, and the experiments in Section 9. An important difference is that we have also added a background order to the decision-making process (compare Eqs. (1) and (6) with [13, Eqs. (1) and (3)]), which is not only a reasonable addition but also crucial for the connection with the coherent extension. This difference affected much of the rest of the paper as well, but the structure of Sections 2 to 6 is still similar to those of the corresponding sections in [13], and the results are closely related.

2. Decision-making with choice functions

We assume that we have an experiment with ℓ possible outcomes whose actual outcome is uncertain. We order the set of all possible outcomes $\mathcal{X}$ as $\{x_1, \ldots, x_\ell\}$, and refer to $\mathcal{X}$ as the state space. Furthermore, we assume that we want to choose between a number of options, the utility of which depends on the outcome of the experiment.

Mathematically, an *option* u is then a function that maps each outcome x in $\mathcal{X}$ to the real-valued utility $u(x)$ that we get when we choose that option and the outcome of the uncertain experiment turns out to be x. The set $\mathcal{V} := \mathbb{R}^{\mathcal{X}}$, i.e. the real vector space of all real-valued maps on $\mathcal{X}$, collects all options and $\mathcal{Q}$ is the set of all non-empty *finite* subsets of $\mathcal{V}$, while we reserve the notation $2^{\mathcal{V}}$ for the power set of $\mathcal{V}$.

[2]This is often just called the natural extension, with 'coherent' being implicit [4, 5]. We here call it the coherent extension—with 'natural' left implicit—to distinguish it from the E-admissible (natural) extension.

Moreover, $\mathcal{Q}_\emptyset \coloneqq \mathcal{Q} \cup \{\emptyset\}$ is the set of all finite subsets of $\mathcal{V}$. Furthermore, for every constant $a \in \mathbb{R}$ we use $a \in \mathcal{V}$ to denote the option that maps every outcome to a.

As a mathematical model for decision-making with options we will use choice functions. Such a choice function is simply a function that, for any given finite set of options $A \in \mathcal{Q}$, selects some non-empty subset of A. More formally, then, a *choice function* C is a map from $\mathcal{Q}$ to $\mathcal{Q}$ that associates a set $C(A) \subseteq A$ with every $A \in \mathcal{Q}$. If $C(A)$ is a singleton consisting of a single option u, this means that u is chosen from A. If $C(A)$ has more than one element, however, we don't take this to mean that all the options in $C(A)$ are chosen, but rather that the options in $A \setminus C(A)$ are rejected and that the model does not contain sufficient information to warrant making a choice between the remaining options in $C(A)$.

We illustrate this with a simple example.

Example 2.1. Let us consider a state space with 3 states: $\mathcal{X} \coloneqq \{1, 2, 3\}$. We identify options with vectors in $\mathbb{R}^3$, where for any $x \in \mathcal{X}$, the x-th component corresponds to the value of the option in x; so for example the option $w_1 \coloneqq (1, -3, 1)$ corresponds to the option that maps 1 to 1, 2 to -3 and 3 to 1.

Suppose now that we have an option set $A \coloneqq \{w_1, w_2, w_3, w_4\}$, with $w_2 \coloneqq (1, 1, -2)$, $w_3 \coloneqq (0, 0, 0)$ and $w_4 \coloneqq (4, -5, -2)$ and that we want to choose from this set. To do this, we will first need some information about the decision-maker's preferences and the decision rule she wants to adopt. $\Diamond$

3. Maximizing expected utility & E-admissibility

Depending on the desired behaviour, various axioms can be imposed on choice functions, leading to different types of choice functions; see for example [10, 11, 12]. In this contribution we focus on choice functions under E-admissibility, as introduced in Section 3.3 further on, and compare these to so-called coherent choice functions from Section 7 onwards. To build up towards these more complicated types of choice functions, we start with some simple ones.

3.1. Strict dominance

A first way to define a choice function is to reject all options for which there is another option whose utility is at least as good as that of the rejected option for all outcomes, and is strictly better for at least one outcome. Mathematically, this can be described by the strict partial order $<$ on $\mathcal{V}$ defined such that for all $v, w \in \mathcal{V}$ we have $v < w$ whenever $v(x) \leq w(x)$ for all $x \in \mathcal{X}$, but $v \neq w$. We call this strict partial order the *background order* and use it to define the choice function $C_<$ as

$$C_<(A) \coloneqq \{u \in A \colon (\forall a \in A) u \nless a\}, \text{ for all } A \in \mathcal{Q}.$$

For any option set $A \in \mathcal{Q}$, the options in $C_<(A)$ are called *admissible* and the others are called *inadmissible* [14]. We illustrate this again with an example.

Example 3.1. In Example 2.1 we wanted to choose from the option set $A \coloneqq \{w_1, w_2, w_3, w_4\}$. By comparing these options pairwise, we find that none of them is strictly dominated by any of the other options. So based on this criterion alone, none of the options can be rejected: $C_<(A) = A$. $\Diamond$

As this example illustrates, the choice function $C_<$ can be very uninformative and does not take into account additional knowledge the decision-maker might have about the uncertain experiment. If we can model her uncertainty, we might be able to do better.

3.2. Maximizing expected utility

When working with a finite state space $\mathcal{X}$, the decision-maker's uncertainty about an experiment is often modelled by means of a probability mass function $p \colon \mathcal{X} \to [0, 1]$, which specifies the probability $p(x)$ of each outcome x in $\mathcal{X}$; we will use

$$\mathbb{P} \coloneqq \Big\{p \colon \mathcal{X} \to [0, 1] \colon \sum_{x \in \mathcal{X}} p(x) = 1\Big\}$$

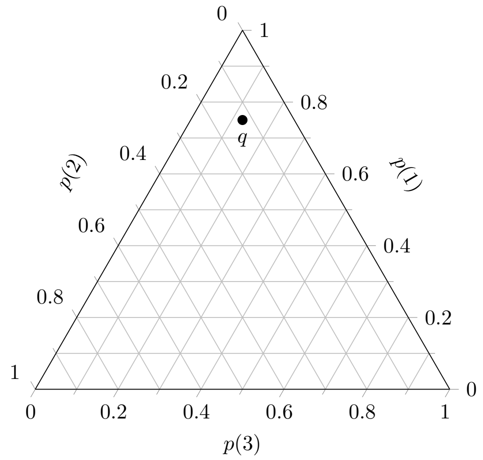


Figure 1: The probability mass function $q = (3/4, 1/8, 1/8) \in \mathbb{P}$ in a ternary diagram.

to denote the set of all such probability mass functions (pmfs) on $\mathcal{X}$. The standard way—see for example [15, Chapter 5]—to then choose between admissible options u proceeds by maximising expected utility with respect to p, where the expected utility of an option $u \in \mathcal{V}$ is given by $\mathrm{E}_p(u) \coloneqq \sum_{x\in\mathcal{X}} u(x)p(x)$. For every probability mass function $p \in \mathbb{P}$, the resulting choice function C_p is defined for all $A \in \mathcal{Q}$ by

$$C_p(A) \coloneqq \{u \in A\colon (\forall a \in A)\mathrm{E}_p(a) \leq \mathrm{E}_p(u) \text{ and } u \not< a\} \tag{1}$$

$$= \{u \in A\colon (\forall a \in A) u \not\prec_p a\}, \tag{2}$$

where for any $u, v \in \mathcal{V}$, we define

$$u \prec_p v \Leftrightarrow [\mathrm{E}_p(u) < \mathrm{E}_p(v) \text{ or } u < v], \tag{3}$$

Alternatively, to clarify the connection with $C_<$, we can also consider the following alternative characterisation:

$$C_p(A) = C_<(A) \cap \hat{C}_p(A), \tag{4}$$

where $\hat{C}_p(A) \coloneqq \{u \in A\colon (\forall a \in A)\mathrm{E}_p(a) \leq \mathrm{E}_p(u)\}$ contains the options in A whose utility with respect to p is maximal.

Example 3.2. Consider the option set A from Example 2.1 and suppose that the decision-maker has a probability mass function $q \in \mathbb{P}$ that assigns probability $3/4$ to the first outcome and probability $1/8$ to each of the other two outcomes. Similar to what we did for options, we will identify such probability mass functions with vectors in $\mathbb{R}^3$. In this particular case, we have $q = (3/4, 1/8, 1/8)$. Note that because of the restrictions imposed by $\mathbb{P}$, probability mass functions reside inside a triangle in $\mathbb{R}^3$ with vertices $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$. This allows us to visualize probability mass functions as points in an equilateral triangle, as illustrated in Fig. 1.

To find $C_q(A)$, we look at the expected utility of the options in A: we find that $\mathrm{E}_q(w_1) = 1/2$, $\mathrm{E}_q(w_2) = 5/8$, $\mathrm{E}_q(w_3) = 0$ and $\mathrm{E}_q(w_4) = 17/8$. Since w_4 is the unique option in A that maximises expected utility with respect to q, we have that $\hat{C}_q(A) = \{w_4\}$. On the other hand, we know from Example 3.1 that $C_<(A) = A$. It therefore follows from Eq. (4) that $C_q(A) = \{w_4\}$. $\lozenge$

It will prove convenient to know that the relation $\prec_p$ is a *preference order* on $\mathcal{V}$, in the following (perhaps somewhat non-standard) sense[3] [16, 11, 4, 17]: a binary relation $\prec$ is a *preference order* on $\mathcal{V}$ whenever for all $u, v, w \in \mathcal{V}$ and $\lambda > 0$,

[3] In many works on decision theory, the term preference order is reserved for a complete and transitive order rather than the definition we use here. We do this for two reasons. First because it aligns with the terminology used in some of our references [4, 11, 14]. Second because writing "strict vector order that extends $<$" is a bit long and cumbersome to write everywhere.

$\prec_0$. $u \not\prec u$, (irreflexivity)

$\prec_1$. if $u \prec v$ and $v \prec w$ then also $u \prec w$, (transitivity)

$\prec_2$. if $u \prec v$ then also $u + w \prec v + w$, (translation invariance)

$\prec_3$. if $u \prec v$ then also $\lambda u \prec \lambda v$, (scaling invariance)

$\prec_4$. if $u < v$ then $u \prec v$. (extends $<$)

Particular combinations of these properties have their own name: a binary relation on the vector space $\mathcal{V}$ is called a strict partial order if it satisfies Properties $\prec_0$ and $\prec_1$ and a strict vector order if it satisfies Properties $\prec_0$ to $\prec_3$. So, a preference order is a strict vector order that extends the strict vector order $<$ on $\mathcal{V}$. We use $\mathbb{O}$ to denote the set of all preference orders on $\mathcal{V}$. The background order $<$ itself is clearly a preference order (it is in fact the smallest—least committal—one) and, as our next result shows, $\prec_p$ is one as well.

Lemma 3.3. For any probability mass function $p \in \mathbb{P}$, the binary relation $\prec_p$ on $\mathcal{V}$ as defined by Eq. (3) is a preference order.

Proof. Property $\prec_0$ follows from the facts that $u \not< u$ and $\mathrm{E}_p(u) \not< \mathrm{E}_p(u)$ for all $u \in \mathcal{V}$. Since E_p is a linear expectation, it is also clearly a 'linear (coherent lower) prevision' in the terminology of Walley [17, §3.2.1 and §3.2.2, p. 128–129]. Hence, by his Theorem 3.8.1 [17, p. 156, 1 and 5], we have that $\prec_p$ is a strict ordering that is 'coherent relative to $\mathcal{V} \times \mathcal{V}$' (in his sense, see [17, §3.7.9, p. 156]), and therefore satisfies Property $\prec_1$ (his R3 in §3.7.6), Property $\prec_2$ (his R5 in §3.7.6), Property $\prec_3$ (his R2 in §3.7.6) and Property $\prec_4$ (his R7 in §3.7.9). □

3.3. E-admissibility

It is, however, not always possible to pin down exact probabilities for all outcomes in $\mathcal{X}$ [17, Chapter 5]. Nevertheless, the decision-maker might still have some knowledge about these probabilities, for example in terms of bounds on the probabilities of some events. Such knowledge gives rise to a set of probability mass functions $\mathcal{P} \subseteq \mathbb{P}$, called a *credal set* [1, Section 1.6.2]. In this context, there need no longer be a unique expected utility and so the decision-maker cannot simply maximise it. Several other decision criteria can then be used instead; Troffaes [14] gives an overview. One criterion that is often favoured is *E-admissibility*: 'choose' those admissible options that maximise expected utility with respect to at least one of the probability mass functions p in $\mathcal{P}$ [1]. The behaviour of choice functions under E-admissibility was first studied for so-called horse lotteries by Seidenfeld [18], Seidenfeld et al. [10], and characterised in terms of axioms for options by De Bock and De Cooman [11] and De Bock [12]. If the credal set $\mathcal{P}$ is non-empty, the corresponding choice function $C^{\mathrm{E}}_{\mathcal{P}}$ is defined by

$$C^{\mathrm{E}}_{\mathcal{P}}(A) \coloneqq \bigcup_{p \in \mathcal{P}} C_p(A) \text{ for all } A \in \mathcal{Q}. \tag{5}$$

It will prove useful to extend this definition to the case $\mathcal{P} = \emptyset$ as well. Eq. (5) then yields that $C^{\mathrm{E}}_{\emptyset}(A) = \emptyset$ for all $A \in \mathcal{Q}$, so $C^{\mathrm{E}}_{\emptyset}$ is not a choice function.

In either case, it follows immediately from Eqs. (2) and (5) that

$$C^{\mathrm{E}}_{\mathcal{P}}(A) = \{u \in A \colon (\exists p \in \mathcal{P})(\forall a \in A) u \not\prec_p a\} \text{ for all } A \in \mathcal{Q}. \tag{6}$$

Alternatively, similarly to what we did in Eq. (4), we can also separate the role of E_p and $<$ in this expression: we first determine the options that are admissible (so not strictly dominated with respect $<$), then determine the undominated ones with respect to E_p for any $p \in \mathcal{P}$, and finally take the intersection of both. For the second part, let us define

$$\begin{aligned}\hat{C}^{\mathrm{E}}_{\mathcal{P}} \colon \mathcal{Q} \to \mathcal{Q}_{\emptyset} \colon A \mapsto \hat{C}^{\mathrm{E}}_{\mathcal{P}}(A) \coloneqq & \bigcup_{p \in \mathcal{P}} \hat{C}_p(A) \\ = & \{u \in A \colon (\exists p \in \mathcal{P})(\forall a \in A) \mathrm{E}_p(a) \leq \mathrm{E}_p(u)\}.\end{aligned}$$

Then by Eqs. (5) and (4), for any $\mathcal{P} \subseteq \mathbb{P}$ and $A \in \mathcal{Q}$,

$$C^{\mathrm{E}}_{\mathcal{P}}(A) = \bigcup_{p\in\mathcal{P}} C_p(A) = \bigcup_{p\in\mathcal{P}} \Big(C_<(A) \cap \hat{C}_p(A)\Big) = C_<(A) \cap \hat{C}^{\mathrm{E}}_{\mathcal{P}}(A). \tag{7}$$

4. Consistent assessments and their E-admissible extensions

In the remainder of this paper, we assume that there is some choice function C that represents the decision-maker's preferences, but that we may not fully know this function. Our partial information about C comes in the form of information about some—so not necessarily all—option sets. More exactly, for some option sets $A \in \mathcal{Q}$, we know that the decision-maker rejects all options in $W \subseteq A$, meaning that $C(A) \subseteq A \setminus W$; this can also be stated as $C(V \cup W) \subseteq V$, with $V \coloneqq A \setminus W$. We will represent such information by an *assessment*: a set $\mathcal{A} \subseteq \mathcal{Q} \times \mathcal{Q}_\emptyset$ of pairs (V, W) of disjoint option sets with the interpretation that, for all $(V, W) \in \mathcal{A}$, the decision-maker definitely rejects the options in W from $V \cup W$. [4] Let us illustrate this again with our running example.

Example 4.1. To choose from the option set A from Example 2.1, we will now assume that we have some information about previous choices. To this end, consider the options $v_1 \coloneqq (-1, 2, -2)$, $v_2 \coloneqq (-2, 2, -1)$, $v_3 \coloneqq (0, 3, -11)$, $v_4 \coloneqq (0, -7, -1)$, $v_5 \coloneqq (0, -2, -1)$ and $v_6 \coloneqq (2, 5, -9)$ and suppose that we are given the following information about the choice function C: v_1 is chosen from $\{v_1, v_2, v_3, v_4\}$—so $C(\{v_1, v_2, v_3, v_4\}) \subseteq \{v_1\}$—but rejected from $\{v_1, v_5, v_6\}$—so $C(\{v_1, v_5, v_6\}) \subseteq \{v_5, v_6\}$. This corresponds to the assessment $\mathcal{A} = \{(\{v_1\}, \{v_2, v_3, v_4\}), (\{v_5, v_6\}, \{v_1\})\}$. ♢

Given such an assessment $\mathcal{A}$, it is natural to ask whether there is a choice function that chooses using E-admissibility that agrees with it, in the sense of the following definition.

Definition 4.2 (Consistency)**.** We call an assessment $\mathcal{A}$ *consistent* (with E-admissibility) if there exists a non-empty set of probability mass functions $\mathcal{P} \subseteq \mathbb{P}$ such that

$$C^{\mathrm{E}}_{\mathcal{P}}(V \cup W) \subseteq V \text{ for all } (V, W) \in \mathcal{A},$$

or equivalently, if $\mathbb{P}(\mathcal{A}) \coloneqq \big\{p \in \mathbb{P} \colon (\forall (V, W) \in \mathcal{A})\, C_p(V \cup W) \subseteq V\big\} \neq \emptyset$.

The equivalence follows from Eq. (5): $C^{\mathrm{E}}_{\mathcal{P}}$ agrees with the assessment $\mathcal{A}$ if and only if $\mathcal{P} \subseteq \mathbb{P}(\mathcal{A})$.

Example 4.3. Consider the assessment $\mathcal{A}$ from Example 4.1. To prove that A is consistent, it suffices to find a single probability mass function that belongs to $\mathbb{P}(\mathcal{A})$. One example is the probability mass function $q = (3/4, 1/8, 1/8)$ from Example 3.2. Indeed, since $\mathrm{E}_q(v_1) = -6/8$, $\mathrm{E}_q(v_2) = -11/8$, $\mathrm{E}_q(v_3) = -1$, $\mathrm{E}_q(v_4) = -1$, $\mathrm{E}_q(v_5) = -3/8$ and $\mathrm{E}_q(v_6) = 1$, we see that $C_q(\{v_1, v_2, v_3, v_4\}) \subseteq \hat{C}_q(\{v_1, v_2, v_3, v_4\}) = \{v_1\}$ and $C_q(\{v_1, v_5, v_6\}) \subseteq \hat{C}_q(\{v_1, v_5, v_6\}) = \{v_6\} \subseteq \{v_5, v_6\}$, which implies that $q \in \mathbb{P}(\mathcal{A})$. Hence, the assessment $\mathcal{A}$ is consistent.

Alternatively, as our next result shows, consistency can also be characterised in terms of $C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}$: it suffices to check for some option set A whether $C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A)$ is empty.

Lemma 4.4. For any $A \in \mathcal{Q}$, an assessment $\mathcal{A}$ is consistent if and only if $C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A) \neq \emptyset$.

Our proof of this result makes use of the following lemma.

[4] We take $V \in \mathcal{Q}$ but $W \in \mathcal{Q}_\emptyset$ for the following reasons: (i) we do not allow $V \cup W$ to be empty because C is only defined for non-empty option sets; (ii) $W = \emptyset$ can happen if we ask an expert to choose from the set V, and he says that he cannot reject anything from it; this corresponds to an uninformative assessment that serves no purpose, but we'll nevertheless allow it; (iii) $V = \emptyset$ means that the expert thinks that all options should be rejected from $V \cup W$; we will see in Lemma 4.4 that this is not consistent with E-admissibility, and therefore exclude this possibility in our assessments.

Lemma 4.5 ([19, I.3 Theorem 3][5])**.** Consider any binary relation $\prec$ on some set S that satisfies Properties $\prec_0$ and $\prec_1$. Then for any finite non-empty set $A \subseteq S$ there is a maximal element $m \in A$ with respect to $\prec$, i.e. an element m such that $m \not\prec a$ for all $a \in A$.

Proof of Lemma 4.4 The implication to the left follows from Eq. (6). To prove the implication to the right, we assume that $\mathbb{P}(\mathcal{A}) \neq \emptyset$ and fix any $p \in \mathbb{P}(\mathcal{A})$. Since we know from Lemma 3.3 that $\prec_p$ is a preference order, it follows from Lemma 4.5 that there is a maximal option $m \in A$ with respect to $\prec_p$ for which $m \not\prec_p a$ for all $a \in A$. Hence, $m \in C_p(A) \neq \emptyset$, which implies that $C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A) \neq \emptyset$ by Eq. (5). □

If an assessment $\mathcal{A}$ is consistent and there is more than one E-admissible choice function that agrees with it, the question remains which one we should use. A careful decision-maker would only want to reject options if this is implied by the assessment. So she wants a most conservative agreeing choice function under E-admissibility: one that rejects the fewest number of options. Since larger credal sets lead to more conservative choice functions, there is a unique most conservative agreeing choice function under E-admissibility, and it is equal to $C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}$. For this reason, for any consistent assessment $\mathcal{A}$, we call $C^{\mathrm{E}}_{\mathcal{A}} \coloneqq C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}$ its *E-admissible (natural) extension*.[6]

So we conclude that checking the consistency of an assessment $\mathcal{A}$, as well as finding the E-admissible extension of a consistent assessment $\mathcal{A}$, both amount to evaluating $C^{\mathrm{E}}_{\mathcal{A}}$. In practice, we will evaluate $C^{\mathrm{E}}_{\mathcal{A}}$ using Eq. (7): we first check if a given option is admissible—whether it is chosen by $C_{<}$—and then check whether it has maximal expected utility for some $p \in \mathbb{P}(\mathcal{A})$—whether it is chosen by $\hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}$. To illustrate this, we take another look at our example.

Example 4.6. Consider again the option set A of Example 2.1 and the assessment $\mathcal{A}$ of Example 4.1. Since we know from Example 4.3 that $\mathcal{A}$ is consistent, its E-admissible extension $C^{\mathrm{E}}_{\mathcal{A}}$ is well-defined, so we can use it to choose a subset $C^{\mathrm{E}}_{\mathcal{A}}(A)$ from the set A. Because we know from Example 3.1 that $C_{<}(A) = A$, it follows from Eq. (7) that $C^{\mathrm{E}}_{\mathcal{A}}(A) = C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})} = \hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A)$. Since $\hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A) = \bigcup_{p\in\mathbb{P}(\mathcal{A})} \hat{C}_p(A)$, we can now proceed to check, for every option $w \in A$, whether there is some probability mass function $p \in \mathbb{P}(\mathcal{A})$ for which $w \in \hat{C}_p(A)$. We have already found in Example 4.3 that $q \in \mathbb{P}(\mathcal{A})$ and for this q we have also found in Example 4.3 that w_4 belongs to $C_q(A)$, which implies that $w_4 \in C^{\mathrm{E}}_{\mathcal{A}}(A)$. But for the other options w_1, w_2 and w_3 we would have to look for other probability mass functions in $\mathbb{P}(\mathcal{A})$ and check whether they maximise expected utility for one of these options. If we find one, that option can be included. However, if an option should be rejected, then we would have to check all probability mass functions in $\mathbb{P}(\mathcal{A})$ to verify that they all reject this option. Since $\mathbb{P}(\mathcal{A})$ typically consists of infinitely many probability mass functions, this is not straightforward. ◊

As this example demonstrates, the main difficulty in evaluating $C^{\mathrm{E}}_{\mathcal{A}}(A)$ lies in the fact that for every $u \in A$, we in principle have to check for every $p \in \mathbb{P}(\mathcal{A})$ whether the option u has maximal expected utility with respect to p. In the following sections we provide a method for performing these checks, using a more practical expression for $\mathbb{P}(\mathcal{A})$.

5. A characterisation of $\mathbb{P}(\mathcal{A})$

In this section we will simplify our expression for $\mathbb{P}(\mathcal{A})$ and rewrite it as a union of convex sets, which will eventually allow us to effectively compute the E-admissible extension of an assessment. The following result is a first simplification of the expression for $\mathbb{P}(\mathcal{A})$.

[5]Note that in this book partial orders are originally defined to be non-strict, but thanks to Lemma 1 in Section I.1 of [19], strict partial orders $\prec$ are connected to non-strict partial orders $\preceq$ by $u \prec v \Leftrightarrow u \preceq v$ and $u \neq v$. Moreover, the definition of maximal elements and the proof of this theorem use strict partial orders.

[6]A more correct name would be the *natural (or most conservative) extension under E-admissibility of the partially specified choice function corresponding to the assessment* $\mathcal{A}$, since it is actually the options that are E-admissible, and it is not exactly the assessment that is extended. However, that would be a bit long and cumbersome for a paper about this topic, so we will stick to the shorter name.

Proposition 5.1. For any assessment $\mathcal{A}$,

$$\mathbb{P}(\mathcal{A}) = \big\{p \in \mathbb{P}\colon (\forall (V,W) \in \mathcal{A})(\forall w \in W)(\exists v \in V)\ w \prec_p v\big\}.$$

Proof. By the definition of $\mathbb{P}(\mathcal{A})$, we have to prove that for all $p \in \mathbb{P}$ and $(V,W) \in \mathcal{A}$ the following statements are equivalent:

$$C_p(V \cup W) \subseteq V \tag{8}$$

and

$$(\forall w \in W)(\exists v \in V)\ w \prec_p v. \tag{9}$$

Take any $p \in \mathbb{P}$ and $(V,W) \in \mathcal{A}$. First we prove that Eq. (8) implies Eq. (9). From Eq. (8) and the fact that V and W are disjoint, it follows that $w \notin C_p(V \cup W)$ for all $w \in W$. This means by definition of C_p that

$$(\forall w \in W)(\exists a \in V \cup W)\ w \prec_p a. \tag{10}$$

We will now show that this implies Eq. (9). Take any option $w \in W$. Let $R \coloneqq \{r \in W\colon w \prec_p r\}$. In the case that R is empty we have by this and Eq. (10) that there is an option $a^* \in V$ such that $w \prec_p a^*$. In the case that R is not empty, it is still finite because it is a subset of W and therefore, by Lemma 4.5, there is some option $w^* \in R$ that is maximal with respect to $\prec_p$. Since $w^* \in R \subseteq W$, we know from Eq. (10) that there is some $a^* \in V \cup W$ such that $w^* \prec_p a^*$. Since $w^* \in R$, we also know that $w \prec_p w^*$ and hence that $w \prec_p a^*$. It is therefore impossible that $a^* \in W$, because this would imply that $a^* \in R$, which is impossible because w^* is maximal in R with respect to $\prec_p$, contradicting $w^* \prec_p a^*$. Hence, it must be that $a^* \in V$. So we have found some a^* in V such that $w \prec_p a^*$. As this holds for any option $w \in W$, we have proved Eq. (9).

Next we prove that Eq. (9) implies Eq. (8). Take any option $w \in W$. Since $V \subseteq V \cup W$, we have from Eq. (9) and the definition of C_p that $w \notin C_p(V \cup W)$. Since this holds for any $w \in W$, it follows that $C_p(V \cup W) \subseteq V$, and this is Eq. (8). $\square$

To rewrite $\mathbb{P}(\mathcal{A})$ as a union of convex sets, we use a similar thought process and notation as in [4, Section 3.2]. First, we take a closer look at the expression for $\mathbb{P}(\mathcal{A})$ in Proposition 5.1. Since we know from Property $\prec_2$ that $w \prec_p v$ is equivalent to $0 \prec_p v - w$, this expression immediately implies that

$$\mathbb{P}(\mathcal{A}) = \{p \in \mathbb{P}\colon (\forall H \in \mathcal{H}_{\mathcal{A}})(\exists h \in H)\ 0 \prec_p h\},$$

with

$$\mathcal{H}_{\mathcal{A}} \coloneqq \{\{v - w\colon v \in V\}\colon (V,W) \in \mathcal{A}, w \in W\}.$$

We call this $\mathcal{H}_{\mathcal{A}}$ the *conjunctive generator*.[7] So we see that $\mathbb{P}(\mathcal{A})$ is a specific instance of a set of probability mass functions of the form

$$\mathbb{P}(\mathcal{H}) \coloneqq \{p \in \mathbb{P}\colon (\forall H \in \mathcal{H})(\exists h \in H)\ 0 \prec_p h\}, \tag{11}$$

with $\mathcal{H} \subseteq \mathcal{Q}_\emptyset$ a set of option sets: $\mathbb{P}(\mathcal{A}) = \mathbb{P}(\mathcal{H}_{\mathcal{A}})$. Let us illustrate this again with our running example.

Example 5.2. For the assessment $\mathcal{A}$ from Example 4.1, we get

$$\begin{aligned}\mathcal{H}_{\mathcal{A}} &= \big\{\{v_1 - v_2\}, \{v_1 - v_3\}, \{v_1 - v_4\}, \{v_6 - v_1, v_5 - v_1\}\big\} \\ &= \big\{\{(1,0,-1)\}, \{(-1,-1,9)\}, \{(-1,9,-1)\}, \{(1,-4,1),(3,3,-7)\}\big\}. \quad \diamond\end{aligned}$$

To rewrite Eq. (11) in a more convenient form, we introduce for any option $h \in \mathcal{V}$ a corresponding set of probability mass functions $\mathbb{P}_h \coloneqq \{p \in \mathbb{P}\colon 0 \prec_p h\}$ and for any option set $H \in \mathcal{Q}_\emptyset$ the set

$$\mathbb{P}[H] \coloneqq \{p \in \mathbb{P}\colon (\exists h \in H)\ 0 \prec_p h\} = \bigcup_{h \in H} \mathbb{P}_h.$$

[7] This name is chosen because of similarities with the conjunctive normal form; see Eq. (12).

Then Eq. (11) can be rewritten as

$$\mathbb{P}(\mathcal{H}) = \bigcap_{H \in \mathcal{H}} \bigcup_{h \in H} \mathbb{P}_h = \bigcap_{H \in \mathcal{H}} \mathbb{P}[H], \tag{12}$$

where we use the convention that the intersection over the empty set is the universal set, so $\mathbb{P}(\emptyset) = \mathbb{P}$.

Next, we transform the intersection of unions in Eq. (12) into a union of intersections. Since every probability mass function p in the set $\mathbb{P}(\mathcal{H})$ corresponds to a preference order $\prec_p$ that prefers at least one option h in each option set $H \in \mathcal{H}$ to zero, we will split the set $\mathbb{P}(\mathcal{H})$ in (possibly overlapping) subsets of probability mass functions, according to which options h in each H they prefer to zero.

To formalize this, for any $\mathcal{H} \subseteq \mathcal{Q}_\emptyset$, we let $\Phi(\mathcal{H})$ be the set of *selection functions* on $\mathcal{H}$: those maps $\phi\colon \mathcal{H} \to \mathcal{V}$ such that $\phi(H) \in H$ for every $H \in \mathcal{H}$. Then

$$\mathcal{G}(\mathcal{H}) \coloneqq \{\{\phi(H)\colon H \in \mathcal{H}\}\colon \phi \in \Phi(\mathcal{H})\} \tag{13}$$

is the set of all sets that can be obtained by selecting one option from each $H \in \mathcal{H}$. The case where $\mathcal{H} = \{H_1, ..., H_m\}$ is finite might make this more intuitive, because then

$$\mathcal{G}(\mathcal{H}) = \{\{h_1, ..., h_m\}\colon (h_1, ..., h_m) \in \times_{j=1}^{m} H_j\}. \tag{14}$$

Other than the finite case, a noteworthy case is $\mathcal{G}(\emptyset) = \{\emptyset\}$. This follows from the fact that if there are no sets to select from, selecting one option from 'each' of these sets yields the empty set. More formally, it follows from the fact that there is a single unique function that maps 'every' element of $\emptyset$ to an element of $\mathcal{V}$. Another noteworthy case is $\mathcal{G}(\{\emptyset\}) = \emptyset$, which follows from the fact that $\Phi(\{\emptyset\}) = \emptyset$ because no function can map the empty set to an element of the empty set.

Since $\mathbb{P}(\mathcal{H})$ consists of all preference orders that prefer at least one $h \in H$ to zero for each $H \in \mathcal{H}$, we now find that

$$\begin{aligned}\mathbb{P}(\mathcal{H}) &= \{p \in \mathbb{P}\colon (\exists G \in \mathcal{G}(\mathcal{H}))(\forall g \in G) 0 \prec_p g\} \\ &= \bigcup_{G \in \mathcal{G}(\mathcal{H})} \bigcap_{g \in G} \mathbb{P}_g = \bigcup_{G \in \mathcal{G}(\mathcal{H})} \mathbf{P}[G],\end{aligned} \tag{15}$$

where, for every set of options $G \in 2^{\mathcal{V}}$,

$$\mathbf{P}[G] \coloneqq \bigcap_{g \in G} \mathbb{P}_g = \{p \in \mathbb{P}\colon (\forall g \in G)\ 0 \prec_p g\}, \tag{16}$$

and here also $\mathbf{P}[\emptyset] = \mathbb{P}$.

So we see from Eq. (15) that $\mathbb{P}(\mathcal{H})$ is a particular instance of a set of preference orders of the form

$$\mathbf{P}(\mathcal{G}) \coloneqq \bigcup_{G \in \mathcal{G}} \mathbf{P}[G] = \{p \in \mathbb{P}\colon (\exists G \in \mathcal{G})(\forall g \in G) 0 \prec_p g\},$$

with $\mathcal{G} \subseteq 2^{\mathcal{V}}$ a set of option sets: $\mathbb{P}(\mathcal{H}) = \mathbf{P}(\mathcal{G}(\mathcal{H}))$. Henceforth, we will refer to any set $\mathcal{G} \subseteq 2^{\mathcal{V}}$ that is to be used for this purpose as a *disjunctive generator*, or simply a generator whenever it is clear from the context what type of generator we are referring to. Moreover, we will call an option set G inside a generator $\mathcal{G}$ a *generator set*. Of particular importance is the generator $\mathcal{G}_\mathcal{A} \coloneqq \mathcal{G}(\mathcal{H}_\mathcal{A})$ because it characterises $\mathbb{P}(\mathcal{A})$.

Lemma 5.3. For any assessment $\mathcal{A}$, $\mathbb{P}(\mathcal{A}) = \bigcup_{G \in \mathcal{G}_\mathcal{A}} \mathbf{P}[G] = \mathbf{P}(\mathcal{G}_\mathcal{A})$, and therefore also $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_\mathcal{A})}$ and $\hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})} = \hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_\mathcal{A})}$.

Proof. This follows from the analysis above; in particular, the first part follows from the fact that $\mathbb{P}(\mathcal{A}) = \mathbb{P}(\mathcal{H}_\mathcal{A})$, Eq. (15) for $\mathcal{H}_\mathcal{A}$ and the definition of $\mathbf{P}(\mathcal{G})$. The second part follows from this and the definition of $C^{\mathrm{E}}_{\mathcal{A}}$. □

Let us illustrate this again with our running example.

Example 5.4. Applying Eq. (14) to the conjunctive generator of Example 5.2, we find that $\mathcal{G}_{\mathcal{A}} = \mathcal{G}(\mathcal{H}_{\mathcal{A}}) = \{G_1, G_2\}$, with

$$\begin{aligned} G_1 &\coloneqq \{v_1 - v_2, v_1 - v_3, v_1 - v_4, v_6 - v_1\} \\ &= \{(1, 0, -1), (-1, -1, 9), (-1, 9, -1), (1, -4, 1)\}, \end{aligned}$$

and

$$\begin{aligned} G_2 &\coloneqq \{v_1 - v_2, v_1 - v_3, v_1 - v_4, v_5 - v_1\} \\ &= \{(1, 0, -1), (-1, -1, 9), (-1, 9, -1), (3, 3, -7)\}. \end{aligned}$$

It therefore follows from Lemma 5.3 that $\mathbb{P}(\mathcal{A}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}}) = \mathbf{P}[G_1] \cup \mathbf{P}[G_2]$. ♢

Now that we have decomposed $\mathbb{P}(\mathcal{A})$ into a union of sets of the form $\mathbf{P}[G]$, let us take a closer look at these sets. From the definition of $\mathbf{P}[G]$ in Eq. (16), we see that they are themselves intersections of sets of the form

$$\mathbb{P}_g = \{p \in \mathbb{P} \colon 0 \prec_p g\} = \{p \in \mathbb{P} \colon 0 < g \text{ or } 0 < \textstyle\sum_{x\in\mathcal{X}} g(x)p(x)\},$$

for $g \in G$. If $g > 0$ then we have that $\mathbb{P}_g = \mathbb{P}$, while in general $\mathbb{P}_g$ is the intersection of $\mathbb{P}$ with the open half-space consisting of all $p \in \mathbb{R}^{\mathcal{X}}$ for which $0 < \sum_{x\in\mathcal{X}} p(x)g(x)$. Since $\mathbb{P}$ and all of these half-spaces are convex, it follows that $\mathbb{P}_g$ is convex for any $g \in \mathbb{R}^{\mathcal{X}}$ and that the intersections $\mathbf{P}[G]$ are convex for any $G \in 2^{\mathcal{V}}$. Moreover, if we let $\mathcal{V}_{>0} \coloneqq \{v \in \mathcal{V} \colon v > 0\}$, it follows for all $G \in 2^{\mathcal{V}}$ that

$$\begin{aligned} \mathbf{P}[G] = \mathbf{P}[G \setminus \mathcal{V}_{>0}] &= \bigcap_{g\in G\setminus\mathcal{V}_{>0}} \mathbb{P}_g \\ &= \left\{p \in \mathbb{P} \colon (\forall g \in G \setminus \mathcal{V}_{>0})\ 0 < \textstyle\sum_{x\in\mathcal{X}} p(x)g(x)\right\}. \end{aligned} \tag{17}$$

We illustrate this visually using our running example.

Example 5.5. The credal set $\mathbb{P}(\mathcal{A}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}}) = \mathbf{P}[G_1] \cup \mathbf{P}[G_2]$ of Example 5.4 is depicted in Fig. 2 as the union of the orange region $\mathbf{P}[G_1]$ and blue region $\mathbf{P}[G_2]$. Note that $\mathbf{P}[G_1]$ and $\mathbf{P}[G_2]$ are convex and the intersection of $\mathbb{P}$ with open half-spaces of the form $\{p \in \mathbb{R}^3 \colon 0 < \sum_{x\in\mathcal{X}} p(x)g(x)\}$ with $g \in G_k \setminus \mathcal{V}_{>0}$. The probability mass function q from Example 4.3 belongs to both $\mathbf{P}[G_1]$ and $\mathbf{P}[G_2]$. Two other probability mass functions that will come in handy later on (in Example 6.2) are $p_1 \coloneqq (9/20, 3/20, 8/20)$ and $p_2 \coloneqq (2/5, 2/5, 1/5)$. The first belongs to $\mathbf{P}[G_1]$ (since $\mathrm{E}_{p_1}(v_1 - v_2) = 1/20 > 0$, $\mathrm{E}_{p_1}(v_1 - v_3) = 3 > 0$, $\mathrm{E}_{p_1}(v_1 - v_4) = 1/2 > 0$ and $\mathrm{E}_{p_1}(v_5 - v_1) = 1/4 > 0$), but not to $\mathbf{P}[G_2]$ (since $\mathrm{E}_{p_1}(v_6 - v_1) = -1 \leq 0$). Similarly, it is easy to verify that p_2 belongs to $\mathbf{P}[G_2]$ but not to $\mathbf{P}[G_1]$. ♢

6. Computing the E-admissible extension of an assessment

In the previous section, we saw that $\mathbb{P}(\mathcal{A}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$ is a special case of a set of preference orders of the form $\mathbf{P}(\mathcal{G})$. For that reason, instead of focussing solely on evaluating $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}$, we will also consider the more general problem of computing the E-admissibility choice function $C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}$ that corresponds to a general (disjunctive) generator $\mathcal{G}$. This will turn out to be convenient for evaluating $C^{\mathrm{E}}_{\mathcal{A}}$ as well, since we will see in Section 8 that $\mathcal{G}_{\mathcal{A}}$ is not the only generator $\mathcal{G}$ for which $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}$; there are often other—simpler—such generators as well.

We start things off with the following simple theorem, which provides a convenient characterisation of when an option is chosen by $C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}$.

Theorem 6.1. Consider an option set $A \in \mathcal{Q}$, an option $u \in A$ and a disjunctive generator $\mathcal{G}$. Then

(i) $u \in C_{<}(A)$ if and only if $0 \not< a - u$ for all $a \in A \setminus \{u\}$;

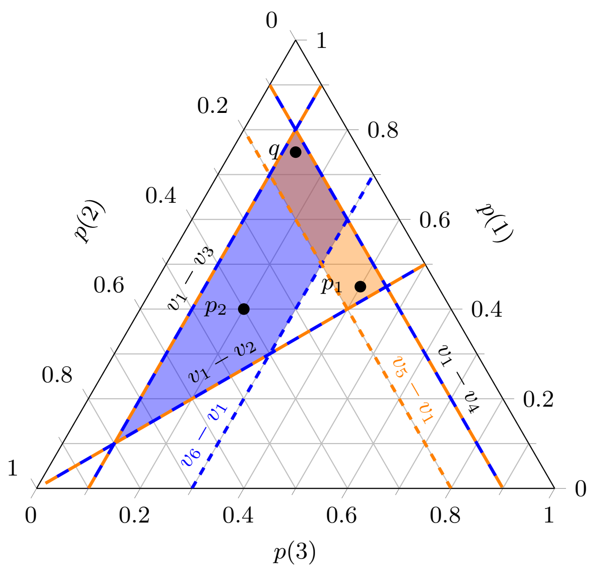


Figure 2: Ternary plot where the credal set $\mathbb{P}(\mathcal{A}) = \mathbf{P}[G_1] \cup \mathbf{P}[G_2]$ consists of those probability mass functions p on $\mathcal{X} = \{1, 2, 3\}$ that belong to the union of the orange region $\mathbf{P}[G_1]$ and the blue region $\mathbf{P}[G_2]$. A dashed line labelled with an option v means that $\mathrm{E}_p(v) = \sum_{x \in \mathcal{X}} v(x)p(x) = 0$ for all p on the line.

(ii) $u \in \hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}(A)$ if and only if there is some $G \in \mathcal{G}$ and $p \in \mathbf{P}[G \setminus \mathcal{V}_{>0}]$ such that, for all $a \in A \setminus \{u\}$, $\mathrm{E}_p(a - u) \leq 0$;

(iii) $u \in C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}(A)$ if and only if $0 \not\prec a - u$ for all $a \in A \setminus \{u\}$ and there is some $G \in \mathcal{G}$ and $p \in \mathbf{P}[G \setminus \mathcal{V}_{>0}]$ such that, for all $a \in A \setminus \{u\}$, $\mathrm{E}_p(a - u) \leq 0$.

Proof. For (i), $u \in C_{<}(A)$ is by definition and Properties $\prec_0$ and $\prec_2$ of $<$ equivalent to $0 \not\prec a - u$ for all $a \in A \setminus \{u\}$.

Let us now look at (ii). Since $\mathbf{P}(\mathcal{G}) = \bigcup_{G \in \mathcal{G}} \mathbf{P}[G]$ and E_p is a linear operator, $u \in \hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}(A)$ if and only if there is some $G \in \mathcal{G}$ and $p \in \mathbf{P}[G]$ such that, for all $a \in A \setminus \{u\}$, $\mathrm{E}_p(a - u) \leq 0$. The equivalence now follows from Eq. (17).

Thanks to Eq. (7), (iii) is a direct consequence of (i) and (ii). $\square$

Let us illustrate the use of Theorem 6.1 by rewriting the problem we left open at the end of Example 4.6.

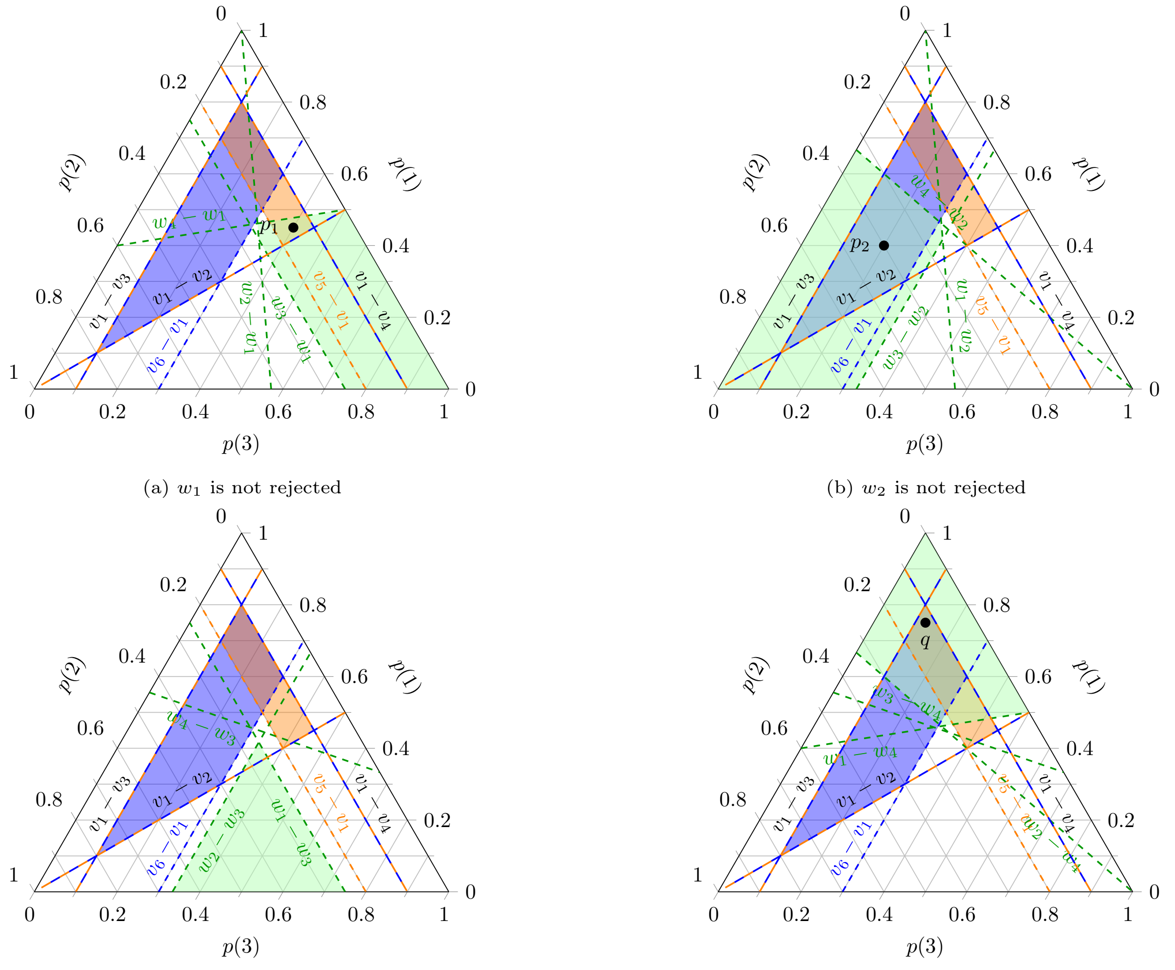
Example 6.2. Recall that in Example 4.6 we set out to evaluate the E-admissible extension $C^{\mathrm{E}}_{\mathcal{A}}$ of the assessment $\mathcal{A}$ in Example 4.1 for the option set $A = \{w_1, w_2, w_3, w_4\}$, with $w_1 = (1, -3, 1)$, $w_2 = (1, 1, -2)$, $w_3 = (0, 0, 0)$ and $w_4 = (4, -5, -2)$. We found that $C^{\mathrm{E}}_{\mathcal{A}}(A) = \hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A)$ and that $w_4 \in \hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A)$, but left the problem of finding out whether w_1, w_2 or w_3 belong to $\hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A)$ open. We now revisit this problem using Theorem 6.1.

We know from Lemma 5.3 that $\hat{C}^{\mathrm{E}}_{\mathbb{P}(\mathcal{A})}(A) = \hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}(A)$ and from Example 5.4 that

$$\begin{aligned}\mathcal{G}_{\mathcal{A}} &= \{G_1, G_2\} \\ &= \{\{(1, 0, -1), (-1, -1, 9), (-1, 9, -1), (1, -4, 1)\}, \\ &\qquad \{(1, 0, -1), (-1, -1, 9), (-1, 9, -1), (3, 3, -7)\}\}.\end{aligned}$$

So we only need to focus on $\hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}$, which corresponds to Item (ii) in Theorem 6.1.

We have already found in Example 5.5 that the probability mass function $p_1 = (9/20, 3/20, 8/20)$ belongs to $\mathbf{P}(\mathcal{G}_1)$. Since $G_1 \setminus \mathcal{V}_{>0} = G_1$ and $\mathrm{E}_{p_1}(w_2 - w_1) = -3/5 \leq 0$, $\mathrm{E}_{p_1}(w_3 - w_1) = -2/5 \leq 0$ and $\mathrm{E}_{p_1}(w_4 - w_1) =$



(a) w_1 is not rejected

(b) w_2 is not rejected

(c) w_3 is rejected

(d) w_4 is not rejected

Figure 3: Ternary plot where the credal set $\mathbb{P}(\mathcal{A})$ consists of those probability mass functions p on $\mathcal{X} = \{1, 2, 3\}$ that belong to the union of the orange region $\mathbf{P}[G_1]$ and blue region $\mathbf{P}[G_2]$. A dashed line labelled with an option v means that $\mathrm{E}_p(v) = 0$ for all p on the line. The green region corresponds to the probability mass functions p for which the options on labelled green lines have a nonpositive expectation.

$-3/20 \leq 0$, it therefore follows from Theorem 6.1 that w_1 is not rejected from A by $\hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_\mathcal{A})}$. We have visualized this in Fig. 3a: the green region contains those probability mass functions for which all of $\mathrm{E}_p(w_2 - w_1)$, $\mathrm{E}_p(w_3 - w_1)$ and $\mathrm{E}_p(w_4 - w_1)$ are non-positive; it is the set of probability mass functions for which w_1 maximises expected utility in A.

That w_2 is not rejected either can be inferred similarly using the probability mass function $p_2 = (2/5, 2/5, 1/5) \in \mathbf{P}(G_1)$ and that w_4 is not rejected can be confirmed analogously using $q = (3/4, 1/8, 1/8) \in \mathbf{P}(G_1) \cap \mathbf{P}(G_2)$. This is illustrated in Figs. 3b and 3d.

For w_3, we observe that the set of probability mass functions for which $\mathrm{E}_p(w_1 - w_3) \leq 0$, $\mathrm{E}_p(w_2 - w_3) \leq 0$ and $\mathrm{E}_p(w_4 - w_3) \leq 0$ corresponds to the green region in Fig. 3c, which has no overlap with the orange region $\mathbf{P}[G_1]$ or the blue region $\mathbf{P}[G_2]$. This implies that w_3 is rejected from A by $\hat{C}_{\mathbf{P}(\mathcal{G}_\mathcal{A})}$.

In conclusion, $C^{\mathrm{E}}_{\mathcal{A}}(A) = \hat{C}^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_\mathcal{A})}(A) = \{w_1, w_2, w_4\}$. ◊

For larger problems, when the graphical approach that we adopted in our running example is no longer feasible, we can translate Theorem 6.1 into an algorithm. The first part of Theorem 6.1, related to the background order $<$, is easy to check. For the second part, we define the function pExists: $\mathcal{Q}_\emptyset \times \mathcal{Q}_\emptyset \to$

{true, false} that, for $F \in \mathcal{Q}_\emptyset$ and $B \in \mathcal{Q}_\emptyset$, returns true if there is some $p \in \mathbf{P}[F]$ such that $\mathrm{E}_p(b) \leq 0$ for all $b \in B$ and false otherwise. Combined, this turns Theorem 6.1 into Algorithm 1. To make sure that this algorithm halts, one should assume that the generator $\mathcal{G}$ is finite, as any rejected option would otherwise correspond to an infinite for-loop over $G \in \mathcal{G}$, and that every $G \in \mathcal{G}$ is finite, as evaluating pExists(G, V) might otherwise take infinitely long. This is in particular always the case for the generator $\mathcal{G}_\mathcal{A}$ corresponding to an assessment $\mathcal{A}$ that is finite: in that case $\mathcal{H}_\mathcal{A}$ and its elements H are clearly finite, which means that $\mathcal{G}_\mathcal{A} = \mathcal{G}(\mathcal{H}_\mathcal{A})$ and its elements G are finite as well because of Eq. (14). The opposite is true as well: if an assessment is infinite, then $\mathcal{G}_\mathcal{A}$ and its elements G will generally be infinite as well. Therefore, in what follows, we will focus on finite assessments and finite generators containing finite option sets.

Algorithm 1 Check for an option set $A \in \mathcal{Q}$ and a disjunctive generator $\mathcal{G}$ if an option $u \in A$ belongs to $C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}(A)$.

$B \leftarrow \{a - u \colon a \in A \setminus \{u\}\}$
for all $b \in B$ **do**
  if $0 < b$ **then**
    return false
for all $G \in \mathcal{G}$ **do**
  if pExists$(G \setminus \mathcal{V}_{>0}, B)$ **then**
    return true ▷ For one of the G's the condition is fulfilled.
return false ▷ When all elements of $G \in \mathcal{G}$ have been checked.

The crucial step in this algorithm is determining pExists$(G \setminus \mathcal{V}_{>0}, B)$. In practice, doing so amounts to solving a linear feasibility problem. In particular, as we will show in Theorem 6.3, it corresponds to a linear feasibility problem of one of the following two types. The primal form looks as follows: for $F = \{f_1, ..., f_m\} \in \mathcal{Q}_\emptyset$ and $B = \{b_1, ..., b_n\} \in \mathcal{Q}_\emptyset$, and recalling that ℓ is the size of the state space $\mathcal{X} = \{x_1, ..., x_\ell\}$,[8]

$$\begin{aligned}
&\text{find } \lambda_1, \dots, \lambda_\ell \in \mathbb{R}, \\
&\text{such that } \textstyle\sum_{i=1}^{\ell} \lambda_i b_k(x_i) \leq 0, && k \in \{1, ..., n\}, \\
&\begin{cases} \sum_{i=1}^{\ell} \lambda_i f_j(x_i) \geq 1, & j \in \{1, ..., m\}, \quad \text{if } F \neq \emptyset \\ \sum_{i=1}^{\ell} \lambda_i \geq 1, & \text{if } F = \emptyset. \end{cases} \\
&\lambda_i \geq 0, && i \in \{1, ..., \ell\},
\end{aligned} \tag{Primal}$$

Alternatively, the dual form looks as follows: for $F = \{f_1, ..., f_m\} \in \mathcal{Q}_\emptyset$ and $B = \{b_1, ..., b_n\} \in \mathcal{Q}_\emptyset$:

$$\begin{aligned}
&\text{find } \lambda_1, \dots, \lambda_{n+m}, \mu \in \mathbb{R}, \\
&\text{such that } \textstyle\sum_{k=1}^{n} \lambda_{m+k} b_k(x_i) - \sum_{j=1}^{m} \lambda_j f_j(x_i) - \mu \geq 0, && i \in \{1, ..., \ell\}, \\
&\textstyle\sum_{j=1}^{m} \lambda_j + \mu \geq 1, \\
&\lambda_k \geq 0, && k \in \{1, ..., n+m\}, \\
&\begin{cases} \mu = 0, & \text{if } F \neq \emptyset, \\ \mu \geq 0, & \text{if } F = \emptyset. \end{cases}
\end{aligned} \tag{Dual}$$

In practice, these feasibility problems can be solved by linear programming.[9] The solver that we'll use in Section 9 uses both the primal and dual form. Our next result relates these feasibility problems to evaluating pExists$(G \setminus \mathcal{V}_{>0}, B)$.

[8] We use the convention that an empty summation is equal to zero.

[9] They can for instance be considered as a standard linear programming problem, by adding the trivial objective function that is zero everywhere. Feeding this into a linear programming software package, the software will announce whether the problem is feasible. For a deeper understanding of how software solves such feasibility problems, we refer to the explanation of initial feasible solutions in [20, Section 5.6].

Theorem 6.3. Consider two option sets $G \in \mathcal{Q}_\emptyset$ and $B \in \mathcal{Q}_\emptyset$. Then the following statements are equivalent:

(i) $\text{pExists}(G \setminus \mathcal{V}_{>0}, B) = \text{true}$;

(ii) (Primal) is feasible for $(G \setminus \mathcal{V}_{>0}, B)$;

(iii) (Dual) is not feasible for $(G \setminus \mathcal{V}_{>0}, B)$.

To prove this result, we first prove the duality of both linear programs.

Lemma 6.4. Consider two option sets $F \in \mathcal{Q}_\emptyset$ and $B \in \mathcal{Q}_\emptyset$. Then (Primal) is feasible for (F, B) if and only if (Dual) is not feasible for (F, B).

Proof. We will use bold letters for vectors and matrices. Note that (Primal) is feasible for $(F, B) = (\{f_1, ..., f_m\}, \{b_1, ..., b_n\})$ if and only if there is some $\boldsymbol{\lambda} \in \mathbb{R}^\ell$ such that $\boldsymbol{\lambda} \geq \mathbf{0}$ and $\mathbf{A}\boldsymbol{\lambda} \leq \mathbf{b}$, with

$$\mathbf{A} \coloneqq \begin{pmatrix} b_1(x_1) & \cdots & b_1(x_\ell) \\ \vdots & & \vdots \\ b_n(x_1) & \cdots & b_n(x_\ell) \\ -f_1(x_1) & \cdots & -f_1(x_\ell) \\ \vdots & & \vdots \\ -f_m(x_1) & \cdots & -f_m(x_\ell) \\ -1 & \cdots & -1 \end{pmatrix} \text{ and } \mathbf{b} \coloneqq \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \\ \vdots \\ -1 \\ -1 \end{pmatrix},$$

where the final row is added only in case $F = \emptyset$. Farkas's Lemma [20, Proposition 6.4.3(ii)] tells us that this is equivalent to the condition that all $\mathbf{y} \in \mathbb{R}^{m+n}$ (or, if $F = \emptyset$, all $\mathbf{y} \in \mathbb{R}^{m+n+1}$) that satisfy $\mathbf{y} \geq \mathbf{0}$ and $\mathbf{y}^\mathrm{T}\mathbf{A} \geq \mathbf{0}^\mathrm{T}$ also satisfy $\mathbf{y}^\mathrm{T}\mathbf{b} \geq 0$. By propositional logic, this is equivalent to the fact that there is no $\mathbf{y} \in \mathbb{R}^{m+n}$ (or, if $F = \emptyset$, no $\mathbf{y} \in \mathbb{R}^{m+n+1}$) that satisfies $\mathbf{y} \geq \mathbf{0}$, $\mathbf{y}^\mathrm{T}\mathbf{A} \geq \mathbf{0}^\mathrm{T}$ and $\mathbf{y}^\mathrm{T}\mathbf{b} < 0$. Since multiplying $\mathbf{y}$ with a positive scalar has no effect on the veracity of these inequalities, this is in turn equivalent to the fact that there is no $\mathbf{y} \in \mathbb{R}^{m+n}$ (or, if $F = \emptyset$, no $\mathbf{y} \in \mathbb{R}^{m+n+1}$) such that $\mathbf{y} \geq \mathbf{0}$, $\mathbf{y}^\mathrm{T}\mathbf{A} \geq \mathbf{0}^\mathrm{T}$ and $\mathbf{y}^\mathrm{T}\mathbf{b} \leq -1$, which holds if and only if (Dual) is not feasible for (F, B). □

Proof of Theorem 6.3 Let us enumerate $G \setminus \mathcal{V}_{>0} = \{g_1, ..., g_m\}$ and $B = \{b_1, ..., b_n\}$. First we prove that (i) implies (ii). By (i), there is some $p \in \mathbf{P}[G \setminus \mathcal{V}_{>0}]$ such that $\mathrm{E}_p(b_k) \leq 0$ for all $k \in \{1, ..., n\}$. Since $p \in \mathbf{P}[G \setminus \mathcal{V}_{>0}]$, it follows from Eq. (17) that $0 < \mathrm{E}_p(g_j)$ for all $j \in \{1, ..., m\}$. Let $\eta \coloneqq \min_{j \in \{1,...,m\}} \mathrm{E}_p(g_j) > 0$ if $G \setminus \mathcal{V}_{>0} \neq \emptyset$ and $\eta = 1$ otherwise. For all $i \in \{1, ..., \ell\}$, let $\lambda_i \coloneqq p(x_i)/\eta \geq 0$. Then on the one hand $\sum_{i=1}^\ell \lambda_i b_k(x_i) = \mathrm{E}_p(b_k)/\eta \leq 0$ for all $k \in \{1, ..., n\}$ because $\mathrm{E}_p(b_k) \leq 0$ and $\eta > 0$. On the other hand, if $G \setminus \mathcal{V}_{>0} \neq \emptyset$, then $\sum_{i=1}^\ell \lambda_i g_j(x_i) = \mathrm{E}_p(g_j)/\eta \geq 1$ for all $j \in \{1, ..., m\}$ because $\mathrm{E}_p(g_j) \geq \eta > 0$, and if $G \setminus \mathcal{V}_{>0} = \emptyset$, then $\sum_{i=1}^\ell \lambda_i = \sum_{i=1}^\ell p(x_i) = 1$. In other words, the real numbers $\lambda_1, \ldots, \lambda_\ell$ satisfy the linear feasibility problem (Primal).

Second we prove that (ii) implies (i). By (ii), there are real numbers $\lambda_1, \ldots, \lambda_\ell$ such that (a) $\lambda_i \geq 0$ for all $i \in \{1, ..., \ell\}$; (b) $\sum_{i=1}^\ell \lambda_i b_k(x_i) \leq 0$ for all $k \in \{1, ..., n\}$; and (c) $\sum_{i=1}^\ell \lambda_i g_j(x_i) \geq 1$ for all $j \in \{1, ..., m\}$ if $G \setminus \mathcal{V}_{>0} \neq \emptyset$ or $\sum_{i=1}^\ell \lambda_i \geq 1$ if $G \setminus \mathcal{V}_{>0} = \emptyset$. Now let $\eta \coloneqq \sum_{i=1}^\ell \lambda_i$. Then $\eta > 0$. This is obvious if $G \setminus \mathcal{V}_{>0} = \emptyset$. If $G \setminus \mathcal{V}_{>0} \neq \emptyset$ this is implied by the constraints in (c). To see why, assume *ex absurdo* that $\eta \leq 0$. Since $\lambda_i \geq 0$ for all $i \in \{1, ..., \ell\}$, this would imply that $\lambda_i = 0$ for all $i \in \{1, ..., \ell\}$. But this would imply that $\sum_{i=1}^\ell \lambda_i g_j(x_i) = 0 \ngeq 1$ for all $j \in \{1, ..., m\}$, which is a contradiction.

Now define $p(x_i) \coloneqq \lambda_i/\eta$ for all $i \in \{1, ..., \ell\}$. Then $p \in \mathbb{P}$, because $\sum_{i=1}^\ell p(x_i) = 1$ and $p(x_i) \geq 0$, and $\mathrm{E}_p(g_j) = \sum_{i=1}^\ell \lambda_i g_j(x_i)/\eta \geq \frac{1}{\eta} > 0$ for all $j \in \{1, ..., m\}$ because $\sum_{i=1}^\ell \lambda_i g_j(x_i) \geq 1$ and $\eta > 0$. This means that $p \in \mathbf{P}[G \setminus \mathcal{V}_{>0}]$, due to Eq. (17). Furthermore, we also have that $\mathrm{E}_p(b_k) = \sum_{i=1}^\ell \lambda_i b_k(x_i)/\eta \leq 0$ for all $k \in \{1, ..., n\}$, because $\sum_{i=1}^\ell \lambda_i b_k(x_i) \leq 0$ and $\eta > 0$, whence $\text{pExists}(G \setminus \mathcal{V}_{>0}, B) = \text{true}$.

The equivalence of (ii) and (iii) follows from Lemma 6.4. □

Now we are able to determine $C^{\mathrm{E}}_{\mathcal{A}}(A)$ for any set $A \in \mathcal{Q}$ and finite assessment $\mathcal{A}$ as follows. Since we know from Lemma 5.3 that $\mathbb{P}(\mathcal{A}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$, it suffices to check for each option $u \in A$ whether $u \in C^{\mathrm{E}}_{\mathcal{A}}(A) = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}$ by applying Algorithm 1 to A and the disjunctive generator $\mathcal{G}_{\mathcal{A}} = \mathcal{G}(\mathcal{H}_{\mathcal{A}})$, which can itself be found by applying Eq. (14) to the conjunctive generator $\mathcal{H}_{\mathcal{A}}$. For any single u, this amounts to checking if the option is strictly dominated and if not, solving a linear feasibility program for subsequent generator sets $G \in \mathcal{G}_{\mathcal{A}}$ until we find a good one or have checked them all, using (Primal) or (Dual), as preferred.

Consistency can also be checked using Algorithm 1, because consistency is equivalent to $C^{\mathrm{E}}_{\mathcal{A}}(A) \neq \emptyset$ for one (and therefore all) $A \in \mathcal{Q}$, by Lemma 4.4. So in practice, since $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}$, consistency can easily be verified beforehand with Algorithm 1, by checking whether $0 \in C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}(\{0\})$. Using (Dual), this reduces to the following lemma.

Lemma 6.5. A finite assessment $\mathcal{A}$ is consistent if and only if there is some $G \in \mathcal{G}_{\mathcal{A}}$, for which the following linear program with $F = G \setminus \mathcal{V}_{>0} = \{f_1, ..., f_m\}$ is not feasible:

$$\begin{aligned} &\text{find } \lambda_1, \dots, \lambda_m \in \mathbb{R}, \\ &\text{such that } \textstyle\sum_{j=1}^{m} \lambda_j f_j(x_i) \leq 0, \quad i \in \{1, ..., \ell\}, \\ &\qquad\qquad \textstyle\sum_{j=1}^{m} \lambda_j \geq 1, \\ &\qquad\qquad \lambda_k \geq 0, \qquad\qquad\quad k \in \{1, ..., m\}. \end{aligned} \tag{Dual$'$}$$

Proof. As explained in the main text that precedes this lemma, consistency is equivalent to the condition that $0 \in C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}(\{0\})$. It follows from Algorithm 1 that this condition is satisfied if and only if pExists$(G \setminus \mathcal{V}_{>0}, \emptyset) =$ true for some $G \in \mathcal{G}_{\mathcal{A}}$. By (i) and (iii) in Theorem 6.3, this is equivalent to the condition that (Dual) is not feasible for some $F = G \setminus \mathcal{V}_{>0}$ with $G \in \mathcal{G}_{\mathcal{A}}$ and $B = \emptyset$. It remains to show that, in its turn, this is equivalent to the condition that (Dual$'$) is not feasible for some $F = G \setminus \mathcal{V}_{>0}$ with $G \in \mathcal{G}_{\mathcal{A}}$. For $F \neq \emptyset$, this equivalence is immediate. For $F = \emptyset$, (Dual) requires us to find some μ such that $-\mu \geq 0$, $\mu \geq 1$ and $\mu \geq 0$. This is clearly impossible, just like (Dual$'$) is not feasible for $F = \emptyset$, as it then contains the requirement $0 \geq 1$. ☐

For the implementation of Algorithm 1, we use the array data structure to store the generators and the option sets inside the generators. For that reason, it could be that a generator or an option set inside a generator is stored multiple times in the array. This is not a problem for two reasons. First, it can be seen immediately from Theorem 6.1 that if there are duplicates inside a generator, then we just do some calculations multiple times, and it has no effect on the end result. Second, if there are duplicates in the linear program then we just get a bigger linear program with the same feasibility. The reason why we allow for duplicates is that we make the assumption that duplicates are rare and that therefore, removing them is not worth the effort.

7. The coherent (natural) extension

The E-admissible (natural) extension is not the only extension that can be derived from an assessment. Another extension that is sometimes considered is the coherent (natural) extension [4, 5]. We will now briefly introduce this coherent extension, and compare it to the E-admissible extension, both theoretically (in this section) and experimentally (in Section 9). For most of this section, Reference [4] will be our guide.

Rather than sets of preference orders derived from probability mass functions, Decadt et al. [4] consider sets of general preference orders and the choice functions they induce. Similar to Eq. (6), given any set of preference orders $\mathcal{O} \subseteq \mathbb{O}$ on $\mathcal{V}$, the choice function—at least if $\mathcal{O}$ is non-empty—that 'chooses' based on these preference orders is defined as

$$C_{\mathcal{O}}(A) \coloneqq \{u \in A \colon (\exists \prec \in \mathcal{O})(\forall a \in A) u \not\prec a\}. \tag{18}$$

A choice function C is then called *coherent* if there is some non-empty set of preference orders $\mathcal{O}$ such that $C = C_{\mathcal{O}}$. In this context, an assessment $\mathcal{A}$ is called consistent (with coherence) if there is some coherent choice function $C_{\mathcal{O}}$ that is compatible with it, in the sense that $C_{\mathcal{O}}(V \cup W) \subseteq V$ for every $(V, W) \in \mathcal{A}$. Because of [4, Proposition 3.6]—similar to Proposition 5.1—the largest set of preference orders $\mathcal{O}$ for which $C_{\mathcal{O}}$ is compatible with $\mathcal{A}$ is

$$\mathbb{O}(\mathcal{A}) := \{\prec \in \mathbb{O} \colon (\forall (V, W) \in \mathcal{A})(\forall w \in W)(\exists v \in V) w \prec v\}. \tag{19}$$

An assessment $\mathcal{A}$ is therefore consistent with coherence if and only if $\mathbb{O}(\mathcal{A}) \neq \emptyset$, and for a consistent assessment, the most conservative coherent choice function that is compatible with it is given by $C_{\mathcal{A}} := C_{\mathbb{O}(\mathcal{A})}$. We call this most conservative coherent choice function the coherent (natural) extension of the assessment $\mathcal{A}$.

For finite assessments, consistency with coherence is equivalent to consistency with E-admissibility.

Lemma 7.1. Consider any finite assessment $\mathcal{A} \subseteq \mathcal{Q} \times \mathcal{Q}_{\emptyset}$. Then $\mathbb{O}(\mathcal{A}) \neq \emptyset$ if and only if $\mathbb{P}(\mathcal{A}) \neq \emptyset$.

Proof. The implication to the left is the easiest to prove, so we will start by proving that. Take any $p \in \mathbb{P}(\mathcal{A}) \neq \emptyset$. Then $\prec_p$ is a preference order because of Lemma 3.3, and belongs to $\mathbb{O}(\mathcal{A})$ because of Proposition 5.1. Hence, $\prec_p \in \mathbb{O}(\mathcal{A}) \neq \emptyset$.

For the reverse implication, we now assume that $\mathbb{O}(\mathcal{A}) \neq \emptyset$. By [4, Proposition 4.3], this means that for some $G = \{g_1, ..., g_m\} \in \mathcal{G}_{\mathcal{A}}$, we have that $\sum_{j=1}^{m} \lambda_j g_j \not\leq 0$ for all $(\lambda_1, ..., \lambda_m) > 0$, in their notation. By the definition of [4, IsFeasible in Section 4.3], this is equivalent to the condition that, for some $G = \{g_1, ..., g_m\} \in \mathcal{G}_{\mathcal{A}}$, 'IsFeasible$(G, 0)$ = false'. By [4, Proposition 4.7], applied for $v = 0$, 'IsFeasible$(G, 0)$ = false' is equivalent to the infeasibility of the following linear program:

$$\begin{aligned}
&\text{find } \lambda_1, \ldots, \lambda_m, \lambda_{m+1} \in \mathbb{R}, \\
&\text{such that } \textstyle\sum_{j=1}^{m} \lambda_j g_j(x_i) \leq 0, && i \in \{1, ..., \ell\}, \\
&\textstyle\sum_{j=1}^{m} \lambda_j \geq 1, \\
&\lambda_k \geq 0, && k \in \{1, ..., m\}, \\
&\lambda_{m+1} \geq 1.
\end{aligned} \tag{Dual$''$}$$

This is exactly the linear program of (Dual$'$) for $F = G$, with the addition of one free variable $\lambda_{m+1} \geq 1$ that does not appear in any other constraints and therefore does not alter the feasibility of the linear program. Hence, (Dual$'$) is not feasible for some $F = G \in \mathcal{G}_{\mathcal{A}}$. By Lemma 6.5, it therefore suffices to prove for any $G \in \mathcal{Q}_{\emptyset}$ that the infeasibility of (Dual$'$) for G implies the infeasibility of (Dual$'$) for $G \setminus \mathcal{V}_{>0}$. This can be seen by contraposition. If (Dual$'$) is feasible for $G \setminus \mathcal{V}_{>0}$, then it is feasible for G as well as we can take the coefficients λ_j of the added options $G \cap \mathcal{V}_{>0}$ to be zero. $\square$

The coherent and E-admissible extensions of a consistent assessment need not be the same though. However, since $\{\prec_p \colon p \in \mathbb{P}(\mathcal{A})\} \subseteq \mathbb{O}(\mathcal{A})$, we do know that $C^{\mathrm{E}}_{\mathcal{A}}(A) = C_{\mathbb{P}(\mathcal{A})}(A) \subseteq C_{\mathbb{O}(\mathcal{A})}(A) = C_{\mathcal{A}}(A)$ for any $A \in \mathcal{Q}$. So, we see that the coherent extension $C_{\mathcal{A}}$ is more conservative than the E-admissible one $C^{\mathrm{E}}_{\mathcal{A}}$.

Just like with the E-admissible extension $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}$, the coherent extension $C_{\mathcal{A}}$ can also be characterised in terms of generators. For any generator $\mathcal{G}$, the corresponding set of preference orders is defined in [4, Eqs. (11) and (12)] as

$$\mathbf{O}(\mathcal{G}) := \{\prec \in \mathbb{O} \colon (\exists G \in \mathcal{G})(\forall g \in G) 0 \prec g\}$$

and, in particular, for $\mathcal{G}_{\mathcal{A}} = \mathcal{G}(\mathcal{H}_{\mathcal{A}})$, we have that $\mathbb{O}(\mathcal{A}) = \mathbf{O}(\mathcal{G}_{\mathcal{A}})$ by [4, Corollary 3.8]. Hence, we see that $C_{\mathcal{A}} = C_{\mathbb{O}(\mathcal{A})} = C_{\mathbf{O}(\mathcal{G}_{\mathcal{A}})}$ is indeed completely characterised by $\mathcal{G}_{\mathcal{A}}$.

8. Simplifications

As mentioned at the beginning of Section 6, $\mathcal{G}_{\mathcal{A}}$ need not be the only generator $\mathcal{G}$ for which $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}$. This will be the case for any $\mathcal{G}$ such that $\mathbf{P}(\mathcal{G}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$. If we can find such a $\mathcal{G}$ that is 'simpler' than $\mathcal{G}_{\mathcal{A}}$, then we can use Algorithm 1 to evaluate $C^{\mathrm{E}}_{\mathcal{A}}$, but more efficiently. This is what we aim to do next.

So what does it mean for a generator to be 'simpler' than another generator? Since we loop over all generator sets in the disjunctive generator $\mathcal{G}$ in Algorithm 1, it is beneficial for $\mathcal{G}$ to contain as few generator sets G as possible and to also have each such G be as small as possible to make the linear feasibility problems smaller.

Decadt et al. [4, Sections 6 and 7] have solved the problem of finding such a simpler generator, be it for the coherent extension instead of the E-admissible one. In particular, they used simplification algorithms to go from an assessment $\mathcal{A}$ to a disjunctive generator $\mathcal{G}^*_{\mathcal{A}}$ that is simpler than $\mathcal{G}_{\mathcal{A}}$ but satisfies $\mathbf{O}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{O}(\mathcal{G}_{\mathcal{A}})$, whereas we require $\mathbf{P}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$. The following result allows us to nevertheless use these methods in our setting as well; it is based on the simple observation that

$$\mathbf{P}(\mathcal{G}) = \{p \in \mathbb{P} \colon \prec_p \in \mathbf{O}(\mathcal{G})\}. \tag{20}$$

Proposition 8.1. Consider two disjunctive generators $\mathcal{G}, \mathcal{G}' \subseteq 2^{\mathcal{V}}$. If $\mathbf{O}(\mathcal{G}) = \mathbf{O}(\mathcal{G}')$, then also $\mathbf{P}(\mathcal{G}) = \mathbf{P}(\mathcal{G}')$.

Proof. By applying Eq. (20) twice, we have that $\mathbf{P}(\mathcal{G}) = \{p \in \mathbb{P} \colon \prec_p \in \mathbf{O}(\mathcal{G})\} = \{p \in \mathbb{P} \colon \prec_p \in \mathbf{O}(\mathcal{G}')\} = \mathbf{P}(\mathcal{G}')$. □

Due to Proposition 8.1, we can use [4, Algorithms 6 and 7] to go from a finite assessment $\mathcal{A}$ to a generator $\mathcal{G}^*_{\mathcal{A}}$ that is simpler than $\mathcal{G}_{\mathcal{A}}$ but nevertheless satisfies $\mathbf{P}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$. This is done in two steps: first use [4, Algorithm 6] to obtain a conjunctive generator $\mathcal{H}^*_{\mathcal{A}}$ for which $\mathbf{O}(\mathcal{G}(\mathcal{H}^*_{\mathcal{A}})) = \mathbf{O}(\mathcal{G}(\mathcal{H}_{\mathcal{A}}))$; then use [4, Algorithm 7] to go from this conjunctive generator $\mathcal{H}^*_{\mathcal{A}}$ to a disjunctive generator $\mathcal{G}^*_{\mathcal{A}}$ for which $\mathbf{O}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{O}(\mathcal{G}(\mathcal{H}^*_{\mathcal{A}}))$. Since then $\mathbf{O}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{O}(\mathcal{G}(\mathcal{H}^*_{\mathcal{A}})) = \mathbf{O}(\mathcal{G}(\mathcal{H}_{\mathcal{A}})) = \mathbf{O}(\mathcal{G}_{\mathcal{A}})$, it indeed follows from Proposition 8.1 that $\mathbf{P}(\mathcal{G}^*_{\mathcal{A}}) = \mathbf{P}(\mathcal{G}_{\mathcal{A}})$. Because of this, if we run Algorithm 1 with the generator $\mathcal{G}^*_{\mathcal{A}}$ instead of $\mathcal{G}_{\mathcal{A}}$, we will obtain the same result because $C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}_{\mathcal{A}})}(A) = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G}^*_{\mathcal{A}})}(A)$. Henceforth, if we write $\mathcal{G}^*_{\mathcal{A}}$, we take this to denote the result of applying [4, Algorithms 6 and 7] to an assessment $\mathcal{A}$.

Instead of using simplification algorithms to directly go from an assessment $\mathcal{A}$ to a generator $\mathcal{G}$—such as $\mathcal{G}^*_{\mathcal{A}}$—that is equivalent to $\mathcal{G}_{\mathcal{A}}$, a more naive approach would be to first explicitly construct the generator $\mathcal{G}_{\mathcal{A}}$ and then simplify it to an equivalent generator $\mathcal{G}$. This would however lead to a combinatorial explosion because constructing $\mathcal{G}_{\mathcal{A}}$ involves taking the Cartesian product of all option sets in $\mathcal{H}_{\mathcal{A}}$ and is therefore not (memory and time) efficient. For more information about how [4, Algorithms 6 and 7] avoid this explosion, we refer to [4, Section(s 5 and) 6].

While these simplifications have been designed in the context of the coherent extension and are therefore mainly intuitive in terms of general preference orders, some of them can also be intuitively understood in terms of credal sets. To that end, an important observation—see [4, Sections 5 and 6]—is that the simplification algorithms [4, Algorithms 5 and 6] only remove options or entire option sets, which means that every option set H in the simplified conjunctive generator $\mathcal{H}^*_{\mathcal{A}}$ will be a subset of some option set H' in the original one and every option set G in the simplified disjunctive generator $\mathcal{G}^*_{\mathcal{A}}$ will be a subset of some G' in the original one. We will use this fact to explain some simplifications in more detail. We focus in particular on [4, Algorithm 6] and also use the fact that $\mathbb{P}(\mathcal{H}_{\mathcal{A}}) = \bigcap_{H \in \mathcal{H}_{\mathcal{A}}} \mathbb{P}[H]$ and $\mathbb{P}[H] = \bigcup_{h \in H} \mathbb{P}_h$.

A first simplification, for example, is that option sets H in $\mathcal{H}_{\mathcal{A}}$ for which there is some $h > 0$ in H are removed. This can be done because $\mathbb{P}_h = \mathbb{P}$ for such $h > 0$ and therefore also $\mathbb{P}[H] = \mathbb{P}$, which implies that $\mathbb{P}(\mathcal{H}_{\mathcal{A}}) = \bigcap_{H' \in \mathcal{H}_{\mathcal{A}}} \mathbb{P}[H'] = \bigcap_{H' \in \mathcal{H}_{\mathcal{A}} \setminus \{H\}} \mathbb{P}[H'] = \mathbb{P}(\mathcal{H}_{\mathcal{A}} \setminus \{H\})$. So the option sets H in the simplified conjunctive generator $\mathcal{H}^*_{\mathcal{A}}$ contain no positive options. Consequently, the disjunctive generator $\mathcal{G}(\mathcal{H}^*_{\mathcal{A}})$ will not contain any positive options, and similarly for any subset of $\mathcal{G}(\mathcal{H}^*_{\mathcal{A}})$ that is smaller—both in number of option sets and options inside each option set—such as the simplified generator $\mathcal{G}^*_{\mathcal{A}}$. This is related to Eq. (17) and makes it so that if we use the simplified generator $\mathcal{G}^*_{\mathcal{A}}$, we do not have to remove the positive options from $G \in \mathcal{G}^*_{\mathcal{A}}$ in line 6 of Algorithm 1, in the sense that $\textsc{pExists}(G \setminus \mathcal{V}_{>0}, B)$ simply becomes $\textsc{pExists}(G, B)$.

A second simplification consists in removing, for any $H \in \mathcal{H}_{\mathcal{A}}$, all $h \in H$ for which $h \leq 0$. This can be done because $\mathbb{P}_h = \emptyset$ for such h and therefore $\mathbb{P}[H] = \bigcup_{h \in H} \mathbb{P}_h = \mathbb{P}[H \setminus \{h\}]$. The option sets H in the simplified conjunctive generator $\mathcal{H}^*_{\mathcal{A}}$ will therefore not contain options $h \leq 0$ and, as a direct consequence, neither will the option sets G in the disjunctive generator $\mathcal{G}(\mathcal{H}^*_{\mathcal{A}})$, or any smaller generator such as the simplified generator $\mathcal{G}^*_{\mathcal{A}}$.

For more details concerning [4, Algorithm 6 and 7] and the many other simplifications they implement, we refer the interested reader to [4, Sections 5 and 6].

9. Experiments

This section is divided into two parts. In the first part we investigate in which cases we should use the simplifications of Section 8 to evaluate the E-admissible extension of a given assessment: when should we use $\mathcal{G}^*_{\mathcal{A}}$ in Algorithm 1 rather than $\mathcal{G}_{\mathcal{A}}$. In the second part we compare the E-admissible extension $C^{\mathrm{E}}_{\mathcal{A}}$ to the coherent extension $C_{\mathcal{A}}$. In particular, we study how Algorithm 1 performs compared to [4, Algorithm 2] and how much more conservative the coherent extension is compared to the E-admissible one. For both parts we have to start from an assessment.

Testing the algorithms is most interesting with consistent assessments because for assessments that are not consistent, the result will always be the empty set, making them rather useless in practice. To get to a consistent assessment, we start from an E-admissible choice function $C^{\mathrm{E}}_{\mathcal{P}}$, apply that choice function to a sequence of random option sets $A_1, ..., A_M$ and use

$$\mathcal{A} \coloneqq \{(V_i, W_i)\colon i \in \{1, ..., M\}\} \coloneqq \{(C^{\mathrm{E}}_{\mathcal{P}}(A_i), A_i \setminus C^{\mathrm{E}}_{\mathcal{P}}(A_i))\colon i \in \{1, ..., M\}\}$$

as the assessment. This way, since $\emptyset \neq \mathcal{P} \subseteq \mathbb{P}(\mathcal{A}_{1:M})$, we obtain a finite assessment that is consistent. In all our experiments, we used $M = 10$. If we want to study the effect of the size of the assessment, we not only consider $\mathcal{A}$ but also consider the partial assessments

$$\mathcal{A}_{1:J} \coloneqq \{(V_i, W_i)\colon i \in \{1, ..., J\}\} = \{(C^{\mathrm{E}}_{\mathcal{P}}(A_i), A_i \setminus C^{\mathrm{E}}_{\mathcal{P}}(A_i))\colon i \in \{1, ..., J\}\},$$

for all $J \in \{1, ..., M\}$. For each assessment $\mathcal{A}_{1:J}$, we consider the generator $\mathcal{G}_{\mathcal{A}_{1:J}}$ and the simplified generator $\mathcal{G}^*_{\mathcal{A}_{1:J}}$. The E-admissible extension $C^{\mathrm{E}}_{\mathcal{A}_{1:J}} = C^{\mathrm{E}}_{\mathbf{P}(\mathcal{G})}$ is evaluated using Algorithm 1 for $\mathcal{G} \in \{\mathcal{G}_{\mathcal{A}_{1:J}}, \mathcal{G}^*_{\mathcal{A}_{1:J}}\}$ and the coherent extension $C_{\mathcal{A}_{1:J}} = C_{\mathbf{O}(\mathcal{G})}$ using [4, Algorithm 2] for $\mathcal{G} = \mathcal{G}^*_{\mathcal{A}_{1:J}}$. The evaluation of the E-admissible extension and the coherent extension is done on 10 separate randomly generated option sets with 8 options each, and we take the average of the results. For a given $\mathcal{A}_{1:J}$, we take the same 10 evaluation option sets, but over the different assessments we took 10 different evaluation option sets.

Table 1 summarizes the parameters, the base point in bold and the values that we will vary around this base point. In this table, the brackets around some of the symbols mean that we will avoid using them, but

Symbolic	Symbol	Parameter values
$\lvert\mathcal{X}\rvert$	(ℓ)	$2, \mathbf{4}, 6, 8$
$\lvert\mathcal{P}\rvert$	(L)	$1, 2, \mathbf{4}, 8$
$\lvert A_i\rvert$ for all i	m	$2, \mathbf{4}, 6, 8$
$\lvert\mathcal{A}_{1:J}\rvert$	(J)	$1, 2, \dots, 9, \mathbf{10}$

Table 1: Parameters and their values with the base point in bold.

rather use their symbolic definition for clarity.

The experiments were run in Julia. We use Gurobi to solve the linear programs, and the experiments were run on the 'Gallade' cluster of the UGent HPC infrastructure. Every evaluation was run on a single core with 7.3 GB of memory on a 2.2 GHz AMD EPYC 7773X processor; for every 'computation'—either preprocessing or evaluation of the choice function—we give a maximum of 60 minutes calculation time. If data points seem to be missing, it is because the computation time of one of the experiments exceeded these limits. If a computation took less than 5 seconds, then we redid the experiment multiple times—up until we had exceeded 5 seconds—and took the average run time in an attempt to get a more accurate 'average' result. For higher computation times, we did not do this because the results were already quite stable.

9.1. Details about the construction of the assessment and option sets

The choice functions that we will use to build our assessment are instances of $C^{\mathrm{E}}_{\mathcal{P}}$, with $\mathcal{P} = \{p_1, ..., p_L\}$ a finite set of probability mass functions because the choice function $C^{\mathrm{E}}_{\mathcal{P}}$ is then easy to evaluate. Since $u \not<_p u$, it follows from Eq. (6) that

$$C^{\mathrm{E}}_{\mathcal{P}}(A_i) = \{u \in A_i\colon (\exists p \in \mathcal{P})(\forall a \in A_i \setminus \{u\}) u \not<_p a\},$$

which can be efficiently evaluated by looping over the L probability mass functions in $\mathcal{P}$: for each $u \in A_i$ we reject it if there is some $a \in A_i \setminus \{u\}$ for which $u < a$ or $\mathrm{E}_p(u) < \mathrm{E}_p(a)$ for some $p \in \mathcal{P}$. To randomly generate $p_1, ..., p_L$, we use [21, Algorithm 1].

The options inside every option set A (the A_i from which we construct the assessment as well as the option sets A for which we evaluate $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}(A)$ and $C_{\mathcal{A}_{1:J}}(A)$) are generated uniformly at random from the unit cube $[0,1]^{|\mathcal{X}|}$. This restriction to $[0,1]^{|\mathcal{X}|}$ does not imply a loss of generality though, as we will explain now. Suppose that we start from an option set of which the elements belong to $\mathbb{R}^{|\mathcal{X}|}$, and not necessarily to the unit cube. We will show that for any such option set $A = \{a_1, ..., a_n\}$ we can always find an equivalent option set A' whose options do belong to the unit cube. To construct A', we consider any two real numbers $t < s$ such that $t \leq a_k \leq s$ for all $k \in \{1, ..., n\}$. Then $A' \coloneqq \{a'_1, ..., a'_n\}$ with $a'_k \coloneqq \frac{a_k - t}{s - t}$ for all $k \in \{1, ..., n\}$ only contains options in the unit cube. Also, every statement of the form $a'_j \prec a'_k$—in particular also for $\prec = \prec_p$—is equivalent to $a_j \prec a_k$ by Properties $\prec_2$ and $\prec_3$ of $\prec$, which, as can be seen from Eq. (19) and Proposition 5.1, is sufficient to guarantee that every assessment $\mathcal{A}$ has an equivalent assessment $\mathcal{A}'$ (in the sense that $C^{\mathrm{E}}_{\mathcal{A}} = C^{\mathrm{E}}_{\mathcal{A}'}$ and $C_{\mathcal{A}} = C_{\mathcal{A}'}$) that only contains options in $[0,1]^{|\mathcal{X}|}$ and, as can be seen from Eqs. (6) and (18), every evaluation of $C^{\mathrm{E}}_{\mathcal{A}}(A)$ and $C_{\mathcal{A}}(A)$ is equivalent to an evaluation of $C^{\mathrm{E}}_{\mathcal{A}}(A')$ and $C_{\mathcal{A}}(A')$ respectively, in the sense that the indices of the chosen options from A match the indices of the chosen options from A'.

9.2. *Effectiveness of the simplifications*

In a first experiment we investigate whether the simplifications of Section 8 have merit in the context of E-admissibility by comparing the time it takes to preprocess an assessment in order to evaluate its E-admissible extension with and without simplifications, as well as the time it takes to evaluate this E-admissible extension with and without simplifications (where, as explained above, we take the average over 10 evaluation option sets with each 8 options). When we evaluate the E-admissible extension without the simplification algorithms of [4], we use a similar method as in [4, Section 7] to run Algorithm 1: we do not construct the disjunctive generator $\mathcal{G}_{\mathcal{A}}$ in memory, but we loop over all combinations of one option from each conjunctive generator set $H \in \mathcal{H}_{\mathcal{A}}$. This is advantageous because the size of the disjunctive generator $|\mathcal{G}_{\mathcal{A}}| = \prod_{H \in \mathcal{H}_{\mathcal{A}}} |H|$ increases exponentially with the number of option sets in the conjunctive generator.

The simplifications should make evaluating the E-admissible extension faster, but at the cost of more preprocessing time. To estimate when the simplifications are worth it, we calculate the number of evaluations n_{BE} of the E-admissible extension at which the total time with and without simplifications break even. Let us therefore use the notation $t^{\times}_{\mathrm{pre}}$ for the time it takes to preprocess the assessment without the simplifications and $t^{\blacktriangle}_{\mathrm{pre}}$ for the time it takes to preprocess the assessment with the simplifications. Similarly, we use $t^{\times}_{\mathrm{ch}}$ for the time it takes to evaluate the E-admissible extension without the simplifications and $t^{\blacktriangle}_{\mathrm{ch}}$ for the time it takes to evaluate the E-admissible extension with the simplifications. Then the number of evaluations n_{BE} at which point both methods break even should satisfy $t^{\blacktriangle}_{\mathrm{pre}} + n_{\mathrm{BE}} t^{\blacktriangle}_{\mathrm{ch}} = t^{\times}_{\mathrm{pre}} + n_{\mathrm{BE}} t^{\times}_{\mathrm{ch}}$. The solution to this equation is given by

$$n = \frac{t^{\blacktriangle}_{\mathrm{pre}} - t^{\times}_{\mathrm{pre}}}{t^{\times}_{\mathrm{ch}} - t^{\blacktriangle}_{\mathrm{ch}}}.$$

In Fig. 4 we plot the time it takes to preprocess and evaluate the E-admissible extension with and without the simplifications, as a function of the length $J = |\mathcal{A}_{1:J}|$ of the assessment, and then also the break-even point n_{BE}. We did 5 separate experiments, each consisting of an assessment and 10 evaluation option sets, which correspond to the different colours in the plots. We furthermore do this for different sizes of the state space $|\mathcal{X}|$, which is also the dimension of the option space $\mathcal{V}$. For $|\mathcal{X}| = 2$, the graphs of the preprocessing time and the graph of the choosing time are almost flat. The choosing times are very close to each other both with and without simplifications, while the preprocessing time is further apart. Therefore, for $|\mathcal{X}| = 2$, the simplifications might not be that useful unless a very high number of evaluations must be made. For $|\mathcal{X}| = 4$ and $|\mathcal{X}| = 6$, we see that the steepness of the curves increases, for both the preprocessing time and the evaluation time. The break-even point n_{BE} becomes lower and the simplifications are more useful. To allow us to fit the results on a single page, we only consider three of the four choices of $|\mathcal{X}|$ in Table 1. We omit the plot for $|\mathcal{X}| = 8$ because the results are similar to those for $|\mathcal{X}| = 6$.

In Fig. 5 we depict the result of an entirely similar experiment, but where we now keep $|\mathcal{X}| = 4$ fixed and instead vary m, the size of the option sets $V \cup W$ in the assessment $\mathcal{A}$. For the base point $m = 4$, this yields the exact same results as for $|\mathcal{X}| = 4$ in Fig. 4. We see that for option sets of size 2, the curves are almost flat and the curves are very close to each other for the choosing time but a bit further apart for the preprocessing time. In this case, the reason is that only binary comparisons are made in the assessment, which implies that $\mathcal{G}_{\mathcal{A}}$ will be either empty or a singleton. The simplifications are therefore not very useful in this case. For bigger option sets, we see that the simplifications do become more useful. We omit the plot for $m = 8$ because it had even fewer points than the one for $m = 6$.

In Fig. 6 we display the result of yet another similar experiment, but where we have now varied $|\mathcal{P}|$, the number of pmfs to create the assessment. For $|\mathcal{P}| = 1$, the simplifications do not take long regardless of the size of the assessment, but the choosing time is also almost the same with and without the simplifications and does not seem to increase much with the size of the assessment. For $|\mathcal{P}| = 4$ and $|\mathcal{P}| = 8$, the simplifications are more useful, but the preprocessing time is also higher. For a higher number of pmfs, the simplifications seem to be less useful, which may be because the credal set is more complex. In this case, we omit the plot for $|\mathcal{P}| = 2$ because it was the least interesting.

Overall, it seems that sometimes the simplifications are beneficial, such as is the case for the bordeaux experiment in the base point, which can be seen in the middle subfigure of Fig. 4. In this case we were unable to compute the E-admissible extension for a bigger assessment without the simplifications. However, in the teal case on the same plot, we are unable to calculate the simplifications for the bigger assessment, while we are able to evaluate the E-admissible extension without the simplifications.

9.3. E-admissible extension vs. natural extension

In this second experiment, we investigate the impact of the size of the assessment on the performance of the E-admissible extension algorithm in comparison with the natural extension algorithm while using the simplifications for constructing the disjunctive generators, and then also on the number of options that are chosen by each extension. We use the simplifications for this comparison because in [4] it was shown that evaluating the coherent extension is usually faster when we use the simplifications; to allow for a fair comparison, we therefore adopt them here as well.

9.3.1. Performance of choosing algorithms

In Fig. 7, we show for different sizes of the state space $|\mathcal{X}|$ a comparison of the time it takes to choose from an option set with 8 options using the E-admissible extension and the coherent extension. For each assessment, the time plotted is the average over the time to choose from 10 different option sets. Note that the time to use the simplifications is not included in this graph, as it is the same for both the natural extension and the E-admissible extension. We see that the E-admissible extension is—averaged over 10 evaluation sets—always faster than the natural extension—averaged over the same 10 evaluation sets—in our experiments, but for high values of $|\mathcal{X}|$ there are fewer data points because the simplifications take too long to process. Similarly, in Fig. 8, we show the same comparison for different sizes m of the option sets in the assessment. We see again that—averaged over 10 evaluation sets—the E-admissible extension is always faster than the natural extension in our experiments and that for high m we have fewer data points as the simplifications take too long to calculate. Then in Fig. 9 we vary the number $|\mathcal{P}|$ of probability mass functions. We see again that the E-admissible extension is faster than the coherent extension.

9.3.2. Difference in number of chosen options

As an extension of the previous experiment, we look at how many options are chosen on average from the 10 evaluation option sets by the E-admissible extension, by the natural extension and also, as a point of reference, by the choice function $C^{\mathrm{E}}_{\mathcal{P}}$ that was used to build the assessments. We look at the same 5 different assessment and for each of the assessments we take the same 10 option sets as in the previous experiment. In Figs. 10 to 12, we look at the dependence of the number of chosen options on J for different dimensions $|\mathcal{X}|$, for different sizes m of option sets in the assessment and for various numbers $|\mathcal{P}|$ of probability mass functions to construct the assessment. In general, we see in these figures that as the assessment gets bigger,

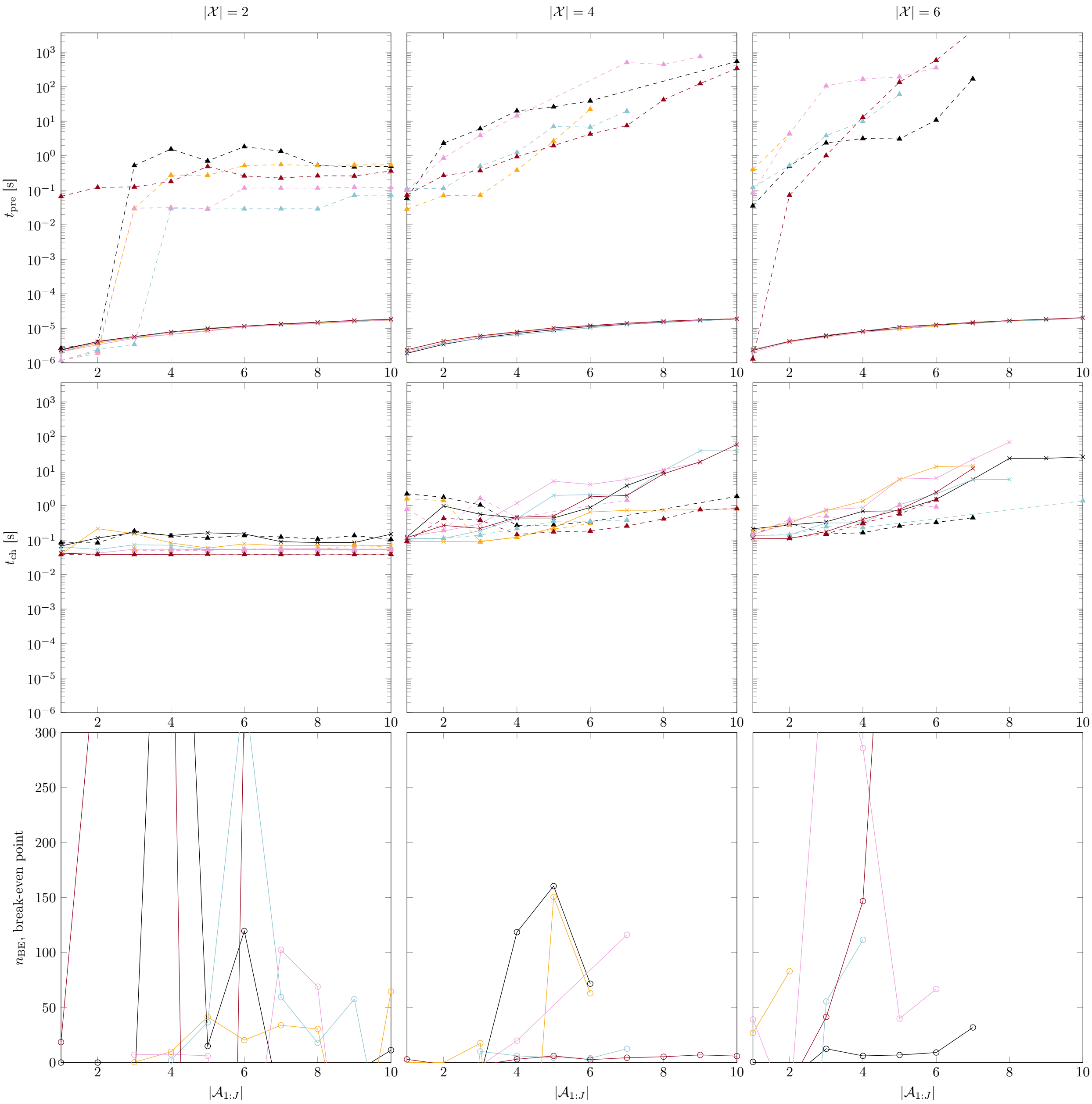


Figure 4: The top and middle row are the time in seconds to preprocess a choice function $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ and to choose from an option set with $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axes are logarithmic and identical. $\times$ is without simplifications, $\blacktriangle$ is with simplifications. The bottom row shows the break-even point for the simplifications. The five colours correspond to five separate experiments. In the columns the dimension $|\mathcal{X}|$ changes.

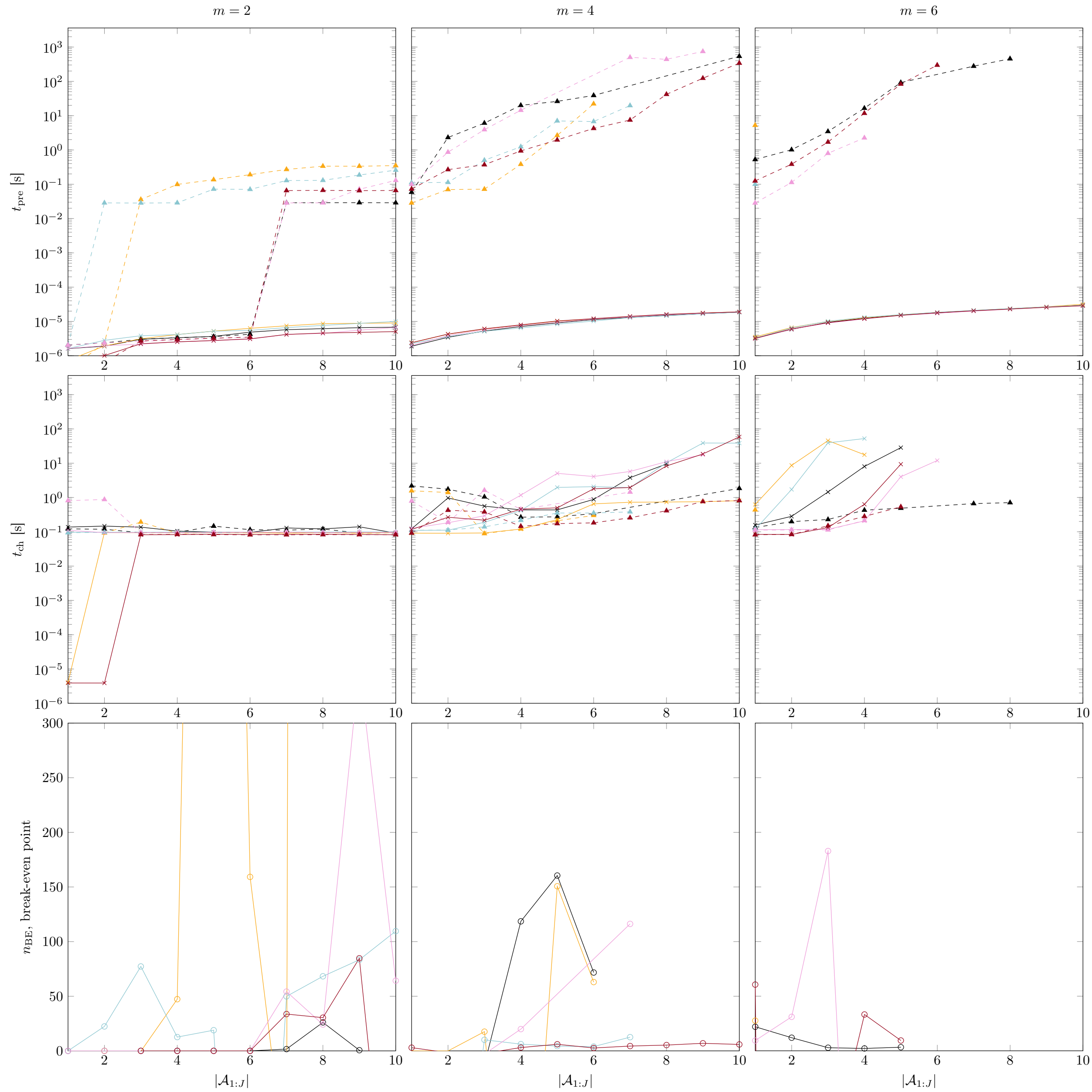


Figure 5: The left and middle column are the time in seconds to preprocess a choice function $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ and to choose from an option set with $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axes are logarithmic and identical. $\times$ is without simplifications, $\blacktriangle$ is with simplifications. The right column shows the break-even point for the simplifications. The five colours correspond to five separate experiments. In the rows the size of the option sets in the assessment changes.

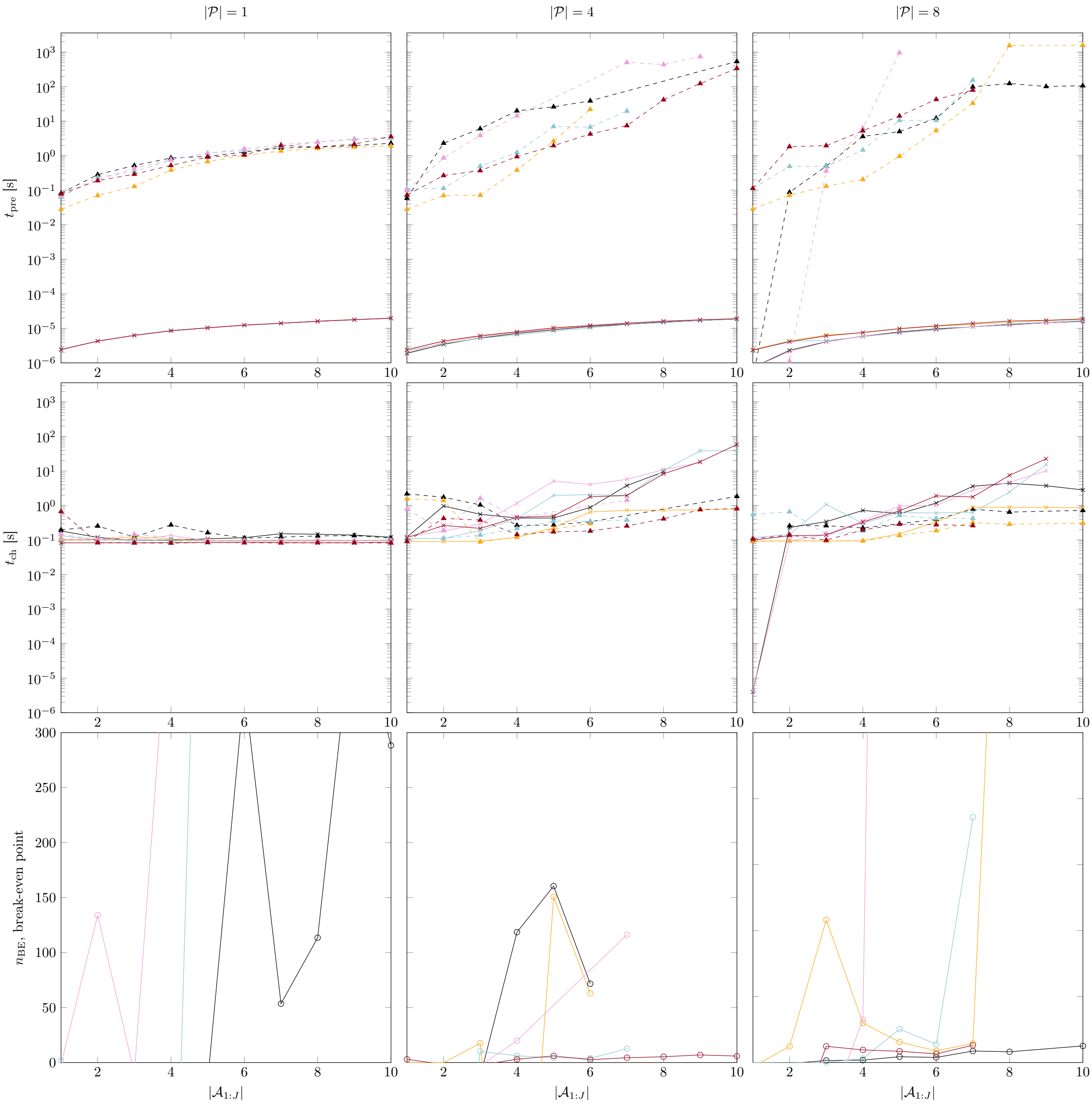


Figure 6: The left and middle column are the time in seconds to preprocess a choice function $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ and to choose from an option set with $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axes are logarithmic and identical. $\times$ is without simplifications, $\blacktriangle$ is with simplifications. The right column shows the break-even point for the simplifications. The five colours correspond to five separate experiments. In the rows the size of $|\mathcal{P}|$ changes.

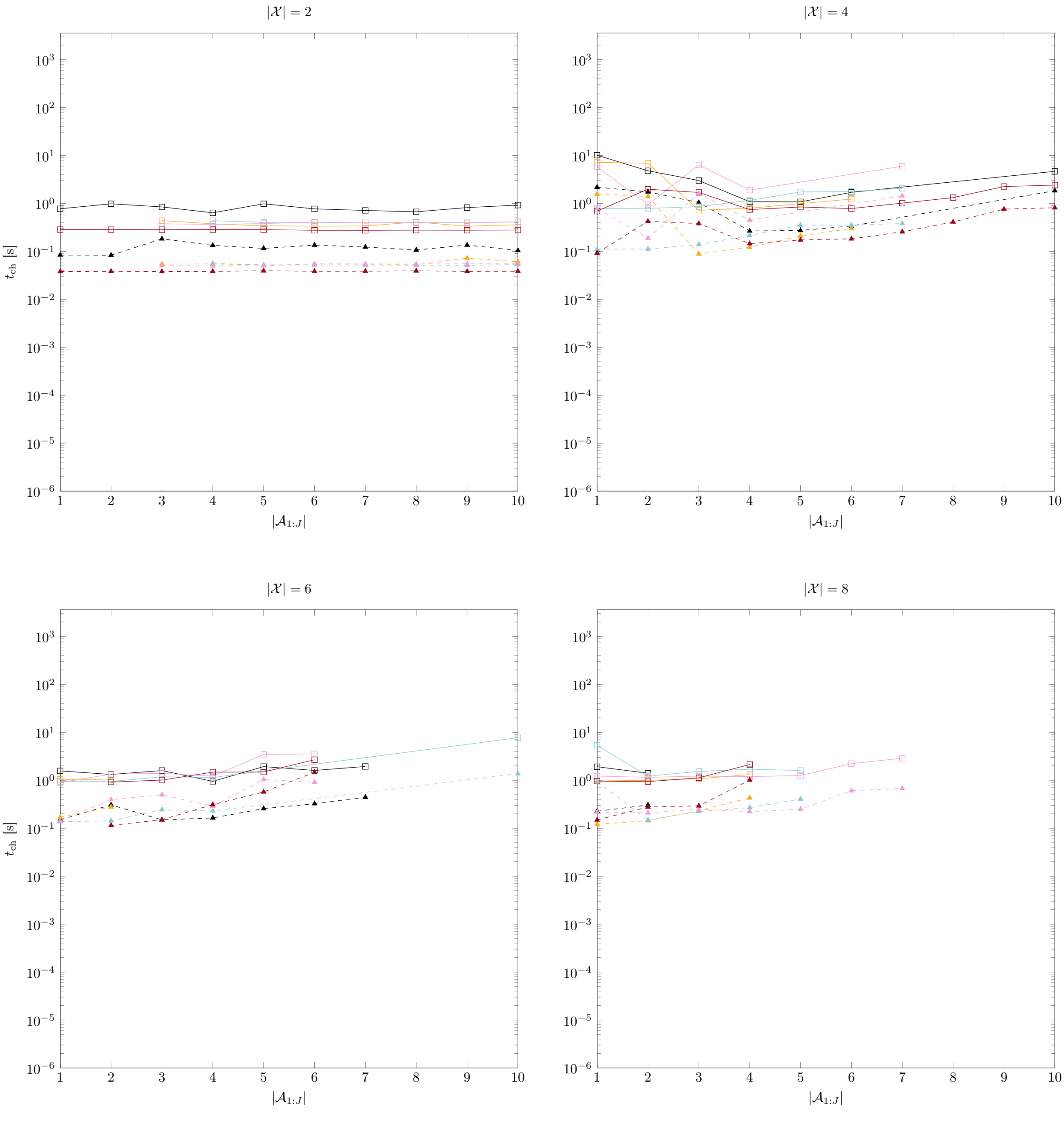


Figure 7: Time in seconds to choose from an option set with either $C_{\mathcal{A}_{1:J}}$ or $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axis is logarithmic, and there are 5 separate experiments each with their own colour and every data point is the average of the time to evaluate the choice function for 10 different option sets. $\square$ is natural extension, $\blacktriangle$ is E-admissibility. Over the different plots we have varied the size of the state space $|\mathcal{X}|$, which is also the dimension of the vector space of options $\mathbb{R}^{|\mathcal{X}|}$.

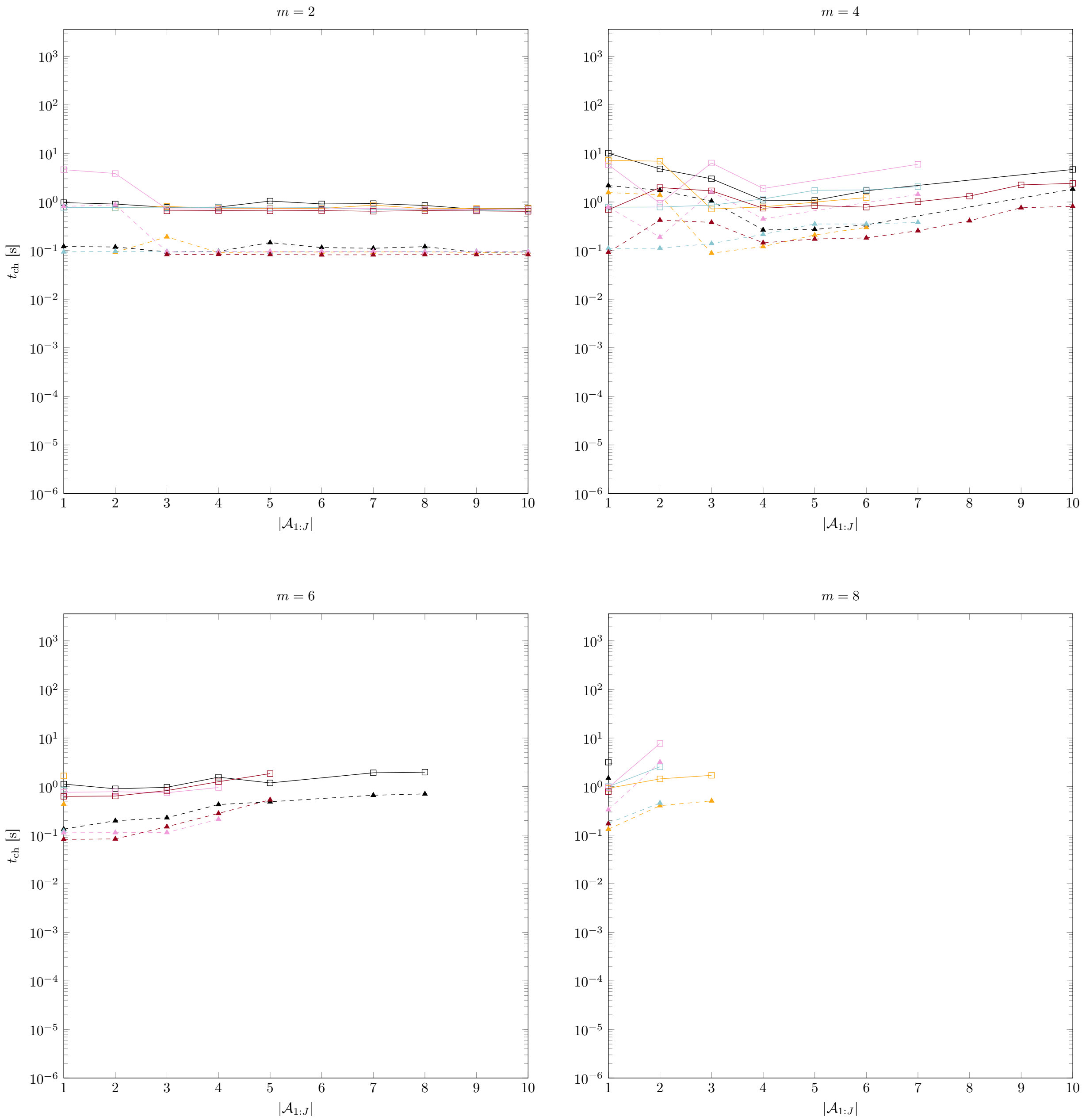


Figure 8: Time in seconds to choose from an option set with either $C_{\mathcal{A}_{1:J}}$ or $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axis is logarithmic, and there are 5 separate experiments each with their own colour and every data point is the average of the time to evaluate the choice function for 10 different option sets. $\square$ is natural extension, $\blacktriangle$ is E-admissibility. Over the different plots we have varied the size m of the option sets in the assessment.

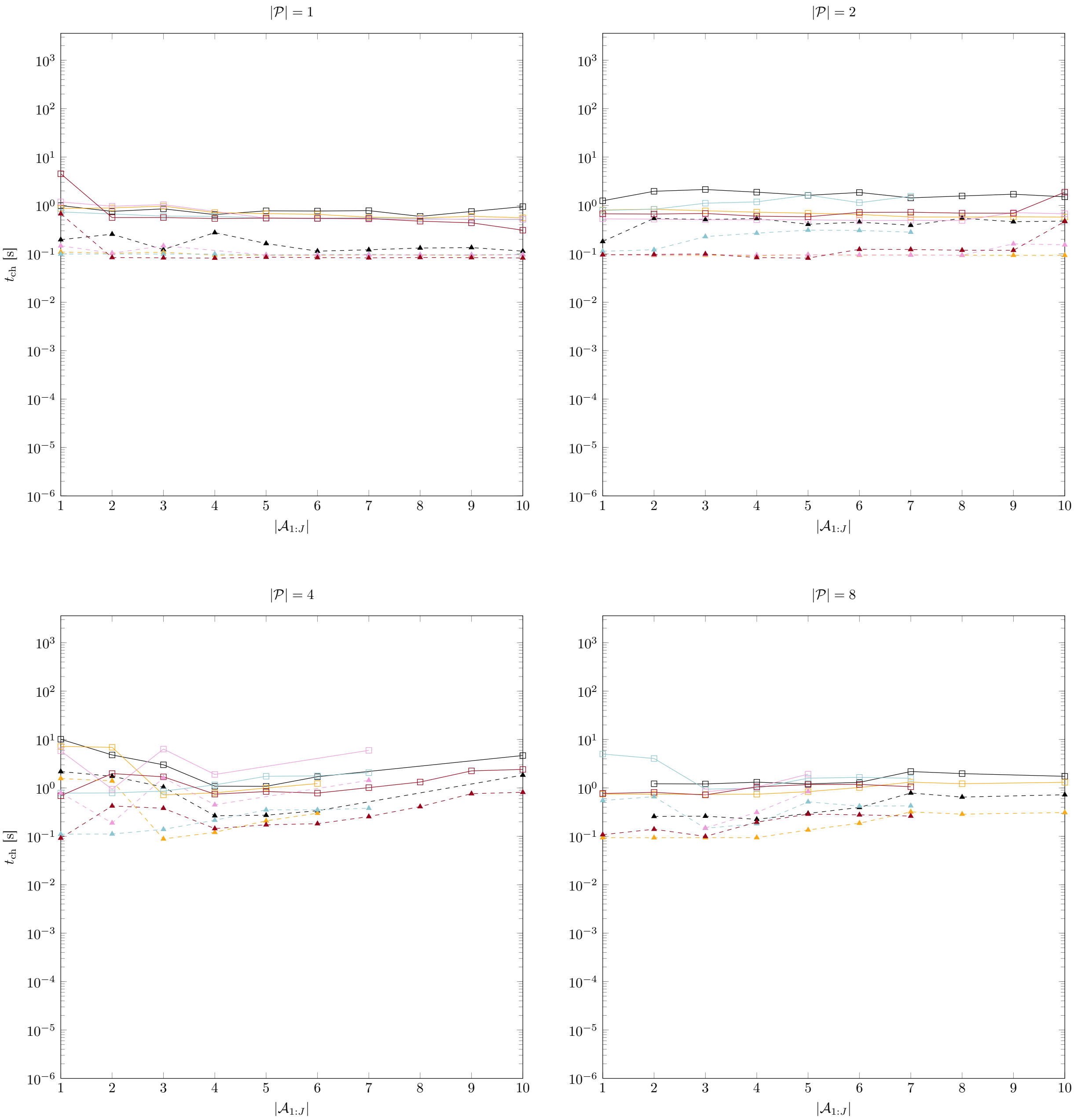


Figure 9: Time in seconds to choose from an option set with either $C_{\mathcal{A}_{1:J}}$ or $C^{\mathrm{E}}_{\mathcal{A}_{1:J}}$ as a function of the number $J = |\mathcal{A}_{1:J}|$ of pairs (V, W) in the assessment. The vertical axis is logarithmic, and there are 5 separate experiments each with their own colour and every data point is the average of the time to evaluate the choice function for 10 different option sets. □ is natural extension, ▲ is E-admissibility. Over the different plots we have varied the number $|\mathcal{P}|$ of probability mass functions to create the assessment.

the number of options chosen by the natural extension and the E-admissible extension slowly gets smaller. We also see that the E-admissible extension is more informative, in the sense that it chooses fewer options than the natural extension. This is not surprising, as it is theoretically guaranteed: for any given assessment, the compatible preference orders of the E-admissible extension are a subset of the compatible preference orders of the coherent extension, whence it can be seen from Eq. (18) that the coherent extension must choose at least as many options as the E-admissible one. What is not obvious, but shown by our experiments, is that the difference between the number of options chosen by the natural extension and the E-admissible extension is, on average, strictly positive and non-negligible, and that it furthermore does not vary too much with the size of the assessment.

In Fig. 10, we see specifically that for lower dimensions we get closer to $C^{\mathrm{E}}_{\mathcal{P}}$ faster. In Fig. 11, we see that for higher sizes of option sets in the assessment, we also get closer to $C^{\mathrm{E}}_{\mathcal{P}}$ faster. Then in Fig. 12, we see that for lower numbers $|\mathcal{P}|$ of probability mass functions used to construct the assessment—that is, if $\mathcal{P}$ becomes more precise—the number of options chosen by $C^{\mathrm{E}}_{\mathcal{P}}$ becomes lower, where for $|\mathcal{P}| = 1$ it is even always 1. The slopes for both the E-admissible extension and the coherent extension are steeper for smaller sizes of $|\mathcal{P}|$.

10. Conclusion

We have studied decision-making under uncertainty while using the E-admissibility criterion and choice functions. In particular, by starting from a finite assessment of past choices, we formulated a method to determine the most conservative choice function under E-admissibility that is compatible with this assessment, through a set of linear feasibility checks.

Compared to earlier algorithmic work on E-admissibility by Utkin and Augustin [8] and Kikuti et al. [9], the main difference is that these authors start from a credal set that is specified by means of linear constraints on probability mass functions (bounds on expectations), which forces their credal sets to be convex polyhedrons. In our approach, on the other hand, we use assessments that specify options that are rejected from given option sets, thereby allowing for credal sets that may be non-convex. In our approach the inequality constraints of Utkin and Augustin [8], Kikuti et al. [9] then essentially correspond to assessments specifying rejections from option sets with two options, but with a key difference: their credal sets are always bounded by non-strict constraints (resulting in closed sets), whereas ours are specified in terms of strict constraints. In our approach, closed credal sets therefore typically require infinite assessments, which our algorithms cannot cope with yet. Therefore, future work could look into which types of infinite assessments can still be handled finitely, so that we get a practical method that can deal with both their constraints and our assessments simultaneously.

We have also related the study of the E-admissible extension to the coherent (natural) extension and have shown that the simplifications of Decadt et al. [4] for the coherent extension can be used for the E-admissible extension as well. For this aspect of our work, future work could consist in developing further simplifications, for example by covering (exactly) the set $\mathbf{P}(\mathcal{G})$ with a minimal number of credal sets of the form $\mathbf{P}[G]$. This is in general an NP-hard problem, but some efficient heuristics do exist [22]. An example where this leads to extra simplifications for $\mathcal{V} = \mathbb{R}^2$ is for the assessment $\mathcal{A} := \{(V, W)\} := \{(\{(2, -1), (-1, 2)\}, \{(0, 0)\})\}$. Then $\mathbb{P}(\mathcal{A}) = \mathbb{P}$, but the simplified generator $\mathcal{G}^*_{\mathcal{A}} = \{\{(2, -1)\}, \{(-1, 2)\}\}$, while $\mathcal{G} = \{\emptyset\}$ is sufficient for $\mathbf{P}(\mathcal{G}) = \mathbb{P}$. The natural extension needs this, as $C_{\mathbb{O}}(V \cup W) = V \cup W$, but $C_{\mathbb{O}(\mathcal{A})}(V \cup W) = V$; while the E-admissible extension has already $C^{\mathrm{E}}_{\mathbb{P}}(V \cup W) = V$.

Another interesting avenue for future work is updating when new information becomes available. Previous computations can be reused, both for the simplifications and for the evaluation of the E-admissible extension and also for the coherent extension.

In our experiments, we have seen that in some cases it is faster to evaluate the E-admissible extension of an assessment with the simplifications of [4], while in other cases it is not. We have also shown that evaluating the E-admissible extension of an assessment is on average faster than evaluating its coherent extension, and yields a smaller (less conservative, more precise) set of choices. The latter is not surprising since we know from Section 7 that the E-admissible extension is always more precise than the coherent

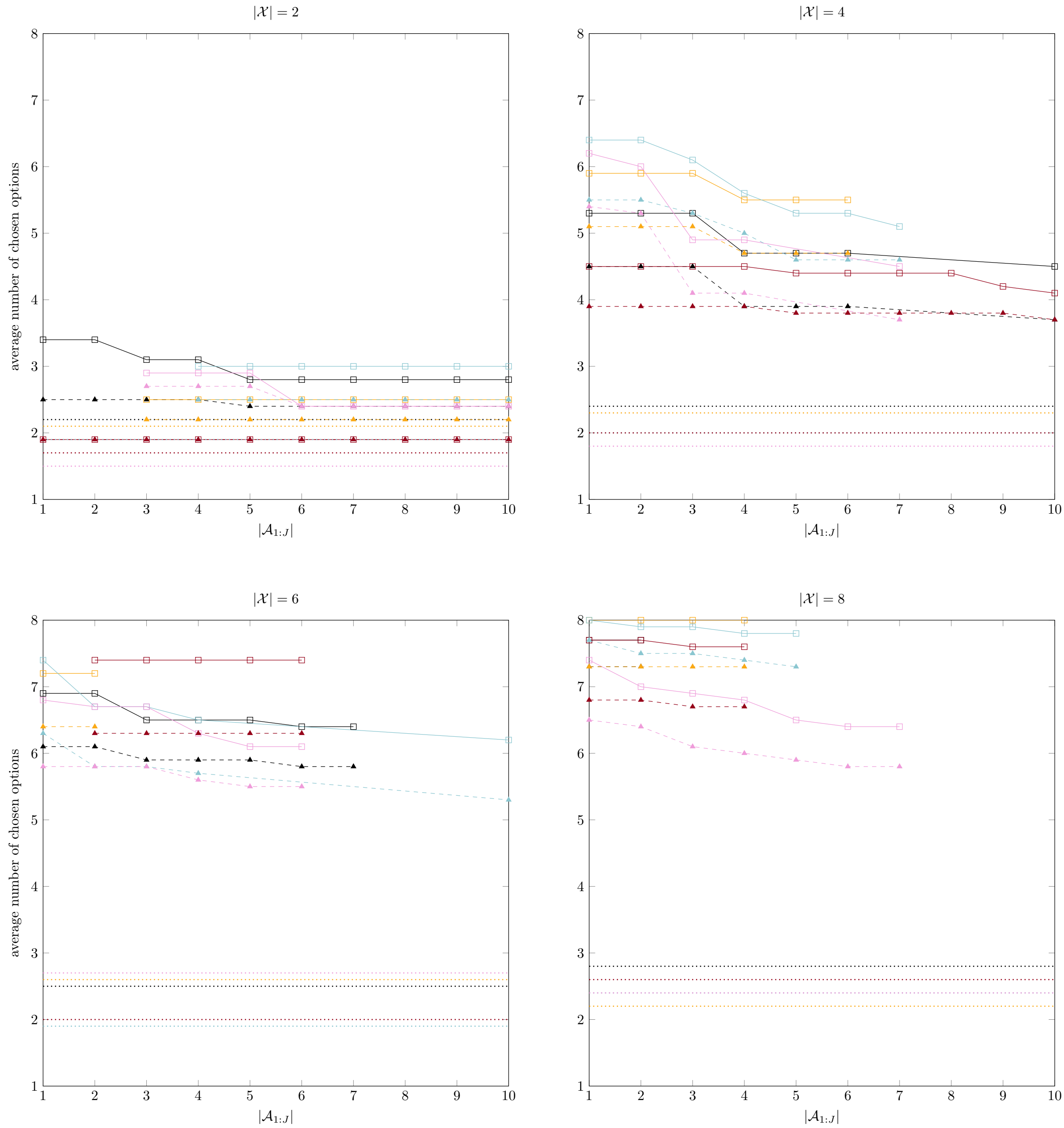


Figure 10: On the horizontal axis, we put the number of pairs (V, W) in the assessment and the vertical axis depicts for the □'s and ▲'s the number of options chosen by the natural extension and the E-admissible extension respectively. The number of options that were chosen by the choice function that was used to construct the assessment are depicted with a dotted line without symbol because they are constant. The 5 colours correspond to 5 separate assessments and each data point depicts the average over 10 random option sets of 8 options to choose from.

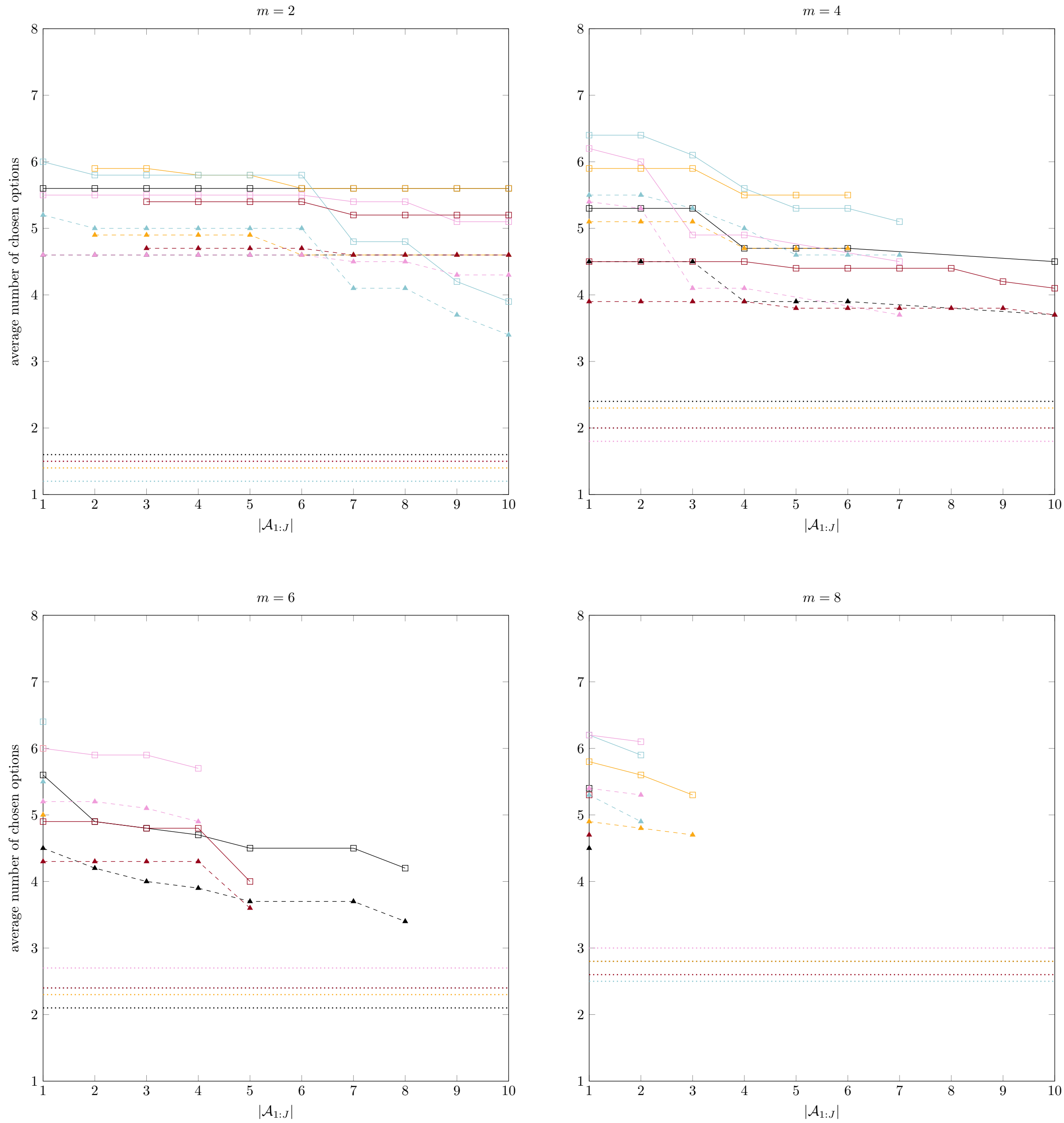


Figure 11: On the horizontal axis, we put the number of pairs (V, W) in the assessment and the vertical axis depicts for the □'s and ▲'s the number of options chosen by the natural extension and the E-admissible extension respectively. The number of options that were chosen by the choice function that was used to construct the assessment are depicted with a dotted line without symbol because they are constant. The 5 colours correspond to 5 separate assessments and each data point depicts the average over 10 random option sets of 8 options to choose from. Over the different plots we have varied the number of options in the option sets of the assessment.

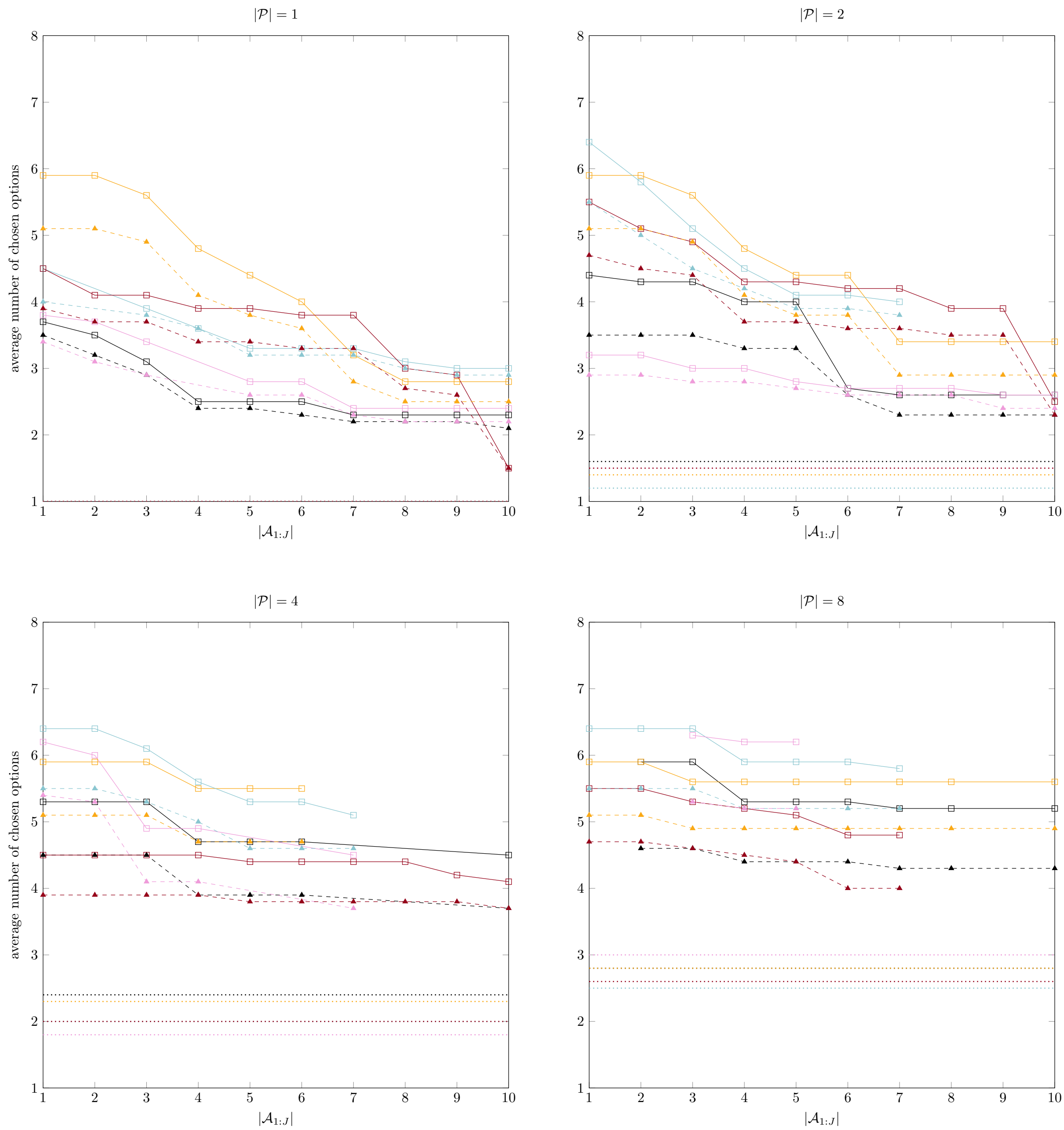


Figure 12: On the horizontal axis, we put the number of pairs (V, W) in the assessment and the vertical axis depicts for the □'s and ▲'s the number of options chosen by the natural extension and the E-admissible extension respectively. The number of options that were chosen by the choice function that was used to construct the assessment are depicted with a dotted line without symbol because they are constant. The 5 colours correspond to 5 separate assessments and each data point depicts the average over 10 random option sets of 8 options to choose from. Over the different plots we have varied the number $|\mathcal{P}|$ of probability mass functions to create the assessment.

extension, but our experiments additionally demonstrate that the difference in how many options we reject is non-negligible. Therefore, we conclude that whenever the assumptions of E-admissibility are justified, it is definitely beneficial, at least from a practical point of view, to use the E-admissible extension instead of the coherent one.

Acknowledgement

The work of all authors was supported by Jasper De Bock's BOF Starting Grant "Rational decision making under uncertainty: a new paradigm based on choice functions", number 01N04819.

References

[1] I. Levi, On indeterminate probabilities, in: Foundations and Applications of Decision Theory, Springer, 1978, pp. 233–261.

[2] I. Levi, The enterprise of knowledge: An essay on knowledge, credal probability, and chance, MIT press, 1980.

[3] I. J. Good, Rational decisions, Journal of the Royal Statistical Society: Series B (Methodological) 14 (1952) 107–114.

[4] A. Decadt, A. Erreygers, J. De Bock, Extending choice assessments to choice functions: An algorithm for computing the natural extension, International Journal of Approximate Reasoning 178 (2025) 109331.

[5] J. De Bock, G. De Cooman, A desirability-based axiomatisation for coherent choice functions, in: Proceedings of SMPS 2018, Springer, 2018, pp. 46–53.

[6] A. K. Sen, Choice functions and revealed preference, The Review of Economic Studies 38 (1971) 307–317.

[7] F. Brandt, P. Harrenstein, Set-rationalizable choice and self-stability, Journal of Economic Theory 146 (2011) 1721–1731.

[8] L. Utkin, T. Augustin, Powerful algorithms for decision making under partial prior information and general ambiguity attitudes., in: Proceedings of the Fourth International Symposium on Imprecise Probabilities and Their Applications, 2005, pp. 349–358.

[9] D. Kikuti, F. G. Cozman, C. P. de Campos, Partially ordered preferences in decision trees: computing strategies with imprecision in probabilities, in: IJCAI workshop on advances in preference handling, 2005, pp. 118–123.

[10] T. Seidenfeld, M. J. Schervish, J. B. Kadane, Coherent choice functions under uncertainty, Synthese 172 (2010) 157–176.

[11] J. De Bock, G. De Cooman, Interpreting, axiomatising and representing coherent choice functions in terms of desirability, in: Proceedings of the Eleventh International Symposium on Imprecise Probabilities: Theories and Applications, PMLR, 2019, pp. 125–134.

[12] J. De Bock, Archimedean choice functions: an axiomatic foundation for imprecise decision making, in: Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems 18th International Conference, Springer, 2020, pp. 195–209.

[13] A. Decadt, A. Erreygers, J. De Bock, G. De Cooman, Decision-making with E-admissibility given a finite assessment of choices, in: Building Bridges between Soft and Statistical Methodologies for Data Science, Springer, 2022, pp. 96–103.

[14] M. C. Troffaes, Decision making under uncertainty using imprecise probabilities, International Journal of Approximate Reasoning 45 (2007) 17–29.

[15] L. J. Savage, The Foundations of Statistics, Courier Corporation, 1972.

[16] R. Frisch, Sur un problème d'économie pure, Grøndahl & søns boktrykkeri, 1926.

[17] P. Walley, Statistical Reasoning with Imprecise Probabilities, Chapman and Hall, 1991.

[18] T. Seidenfeld, Decision theory without "independence" or without "ordering": What is the difference?, Economics & Philosophy 4 (1988) 267–290.

[19] G. Birkhoff, Lattice theory, third ed., American Mathematical Society, 1940.

[20] J. Matoušek, B. Gärtner, Understanding and Using Linear Programming, Universitext, Springer, 2006.

[21] N. Nakharutai, M. C. M. Troffaes, C. C. Caiado, Improved linear programming methods for checking avoiding sure loss, International Journal of Approximate Reasoning 101 (2018) 293–310.

[22] S. J. Eidenbenz, P. Widmayer, An approximation algorithm for minimum convex cover with logarithmic performance guarantee, SIAM Journal on Computing 32 (2003) 654–670.

Appendix: Table of Notation

Symbol	Page	Meaning
$\mathcal{X}$	2	State space: finite set of all outcomes of an experiment
$\mathbb{R}$	2	The real numbers
$\mathcal{V}$	2	$\mathbb{R}^{\mathcal{X}}$, the vector space containing all options
$\mathcal{Q}$	2	The set of all non-empty finite subsets of $\mathcal{V}$
$\mathcal{Q}_\emptyset$	3	$\mathcal{Q} \cup \{\emptyset\}$
$<$	3	background order on $\mathcal{V}$: $u < v \Leftrightarrow ((\forall x \in \mathcal{X})\, u(x) \leq v(x)$ and $(\exists x \in \mathcal{X})\, u(x) < v(x))$
$\mathbb{P}$	3	Contains all probability mass functions on $\mathcal{X}$
E_p	4	Expectation operator induced by the probability mass function p
C_p	4	Choice function that chooses based on maximising expected utility for a probability mass function p
$\prec$	4	Preference order; strict vector order on $\mathcal{V}$ that extends $<$
$\mathbb{O}$; $\mathbb{O}(\mathcal{A})$	5; 16	Contains all preference orders on $\mathcal{V}$; maps an assessment $\mathcal{A}$ to a set of all preference orders that agree with $\mathcal{A}$
$\mathcal{A}$	6	An assessment, subset of $\mathcal{Q} \times \mathcal{Q}_\emptyset$
$C^{\mathrm{E}}_{\mathcal{A}}$	6	E-admissible extension of an assessment $\mathcal{A}$
$\mathcal{H}$	8	Conjunctive generator, subset of $2^{\mathcal{V}}$
$\mathcal{H}_{\mathcal{A}}$	8	Conjunctive generator corresponding to a given assessment $\mathcal{A}$
$\mathbb{P}(\mathcal{A})$; $\mathbb{P}(\mathcal{H})$	6;8	maps an assessment $\mathcal{A}$ or a conjunctive generator $\mathcal{H}$ to the largest credal set that agrees with them
$\mathbb{P}[H]$	8	maps a conjunctive generator set H to the largest credal set that agrees with H
$\mathbb{P}_h$	8	The credal set of all probability mass functions for which h has positive expectation
$\mathcal{G}$	9	Disjunctive generator
$\mathcal{G}_{\mathcal{A}}$	9	Disjunctive generator corresponding to a given assessment $\mathcal{A}$
$\mathbf{P}(\mathcal{G})$; $\mathbf{P}[G]$	9	Maps a disjunctive generator $\mathcal{G}$ or a disjunctive generator set G to the largest credal set that agrees with them

pExists(A, B)	13	Returns true if a probability p exists whose expectation operator maps all options in A to positive real numbers and all option in B all nonpositive real numbers, and false otherwise
$\mathbf{O}(\mathcal{G})$	16	Maps a disjunctive generator $\mathcal{G}$ to set of preference orders that agree with $\mathcal{G}$
$\mathcal{G}^*_{\mathcal{A}}$	17	Simplified disjunctive generator for a given assessment $\mathcal{A}$
L	18	Number of probability mass functions in the experiments
$\times$	19	Cartesian product of sets, in experiments used to mean 'without simplifications'
$\blacktriangle$	19	in experiments used to mean 'with simplifications'