An Axiomatic Framework for Influence Diagram Computation with Partially Ordered Preferences

Nic Wilson[a], Radu Marinescu[b]

[a]*Insight Centre for Data Analytics, University College Cork, Ireland*
[b]*IBM Research, Ireland*

Abstract

This paper presents an axiomatic framework for influence diagram computation, which allows reasoning with partially ordered values of utility. We show how an algorithm based on sequential variable elimination can be used to compute the set of maximal values of expected utility (up to an equivalence relation). Formalisms subsumed by the framework include decision making under uncertainty based on multi-objective utility, or on interval-valued utilities, as well as a more qualitative decision theory based on order of magnitude probabilities and utilities. Consequently, we also introduce the order of magnitude influence diagram to model and solve partially specified sequential decision problems when only qualitative (or imprecise) information is available.

Keywords: Influence diagrams, graphical models, Bayesian networks, variable elimination, preferences, uncertainty, utility, optimization

1. Introduction

Influence diagrams have been widely used for the past three decades as a graphical model to formulate and solve decision problems under uncertainty. The standard formulation of an influence diagram consists of two types of information: *qualitative information* that defines the structure of the problem and *quantitative information* (also known as parametric structure) that, together with the former, defines the model. The qualitative information includes the set of (discrete) chance variables, where the outcome is determined randomly based on the

Email addresses: `nic.wilson@insight-centre.org` (Nic Wilson), `radu.marinescu@ie.ibm.com` (Radu Marinescu)

values assigned to other variables, describing the set of possible world configurations, the set of decision variables, which the decision maker can choose the value of, based on observations on some other variables, as well as the dependencies between the two sets of variables. The parametric structure is composed of the conditional probability distributions associated with each of the chance variables (thus representing uncertainty like in Bayesian networks), as well as a collection of utility functions, whose sum describes the overall value of an outcome, and thus is used to represent the preferences of the decision maker. A policy defines which actions to take for each decision variable, given the available information, and has a corresponding expected utility. The solution to an influence diagram is an optimal policy that maximizes the expected utility and therefore depends on both types of qualitative and quantitative information.

In general, actions can lead to many different kinds of consequences, for example, financial gain/loss, risk to health, effect on the environment or gain/loss to reputation. It may not be possible to map the various potential consequences of a set of actions to the same scale of utility in a way that avoids making essentially arbitrary choices. It is thus natural to consider in this case notions of multi-attribute/objective utility (including imprecise tradeoffs between utility objectives), where utility values are only partially ordered. Quite often, we may have precise knowledge of the qualitative information but only very rough (or imprecise) estimates of the quantitative parameters. In such cases, the standard solution techniques cannot be applied directly, unless the missing or imprecise information is accounted for.

1.1. Contributions

In this paper, we consider decision making under uncertainty using influence diagrams, but where we allow more general notions of uncertainty than probability, and more general notions of utility functions, which, in particular, allow utility values to be only partially ordered. We next highlight the major contributions of the paper, as follows:

- We construct an axiomatic framework, listing properties of a formalism that allow maximal (generalized) expected utility to be computed by sequential elimination of all the variables.
- We prove formally that the set of utility values computed by a sequential variable elimination algorithm within this framework is *equivalent* to the set of maximal values of expected utility.

- We discuss in detail and provide numerical experiments for order of magnitude influence diagrams, a formalism that can be used to model and solve partially specified sequential decision making problems when only qualitative (or imprecise) information is available.

In general terms, variable elimination algorithms can be viewed as follows. We have a collection Θ of functions, where each function in Θ only involves a small number of variables. In the case of influence diagram computation, Θ contains both probability functions and utility functions. Θ is used as a compact representation (or decomposition) of a function $\bigotimes \Theta$ on all the variables, equaling a combination of all the functions in Θ. For example, in a Bayesian network, Θ consists of a collection of conditional probability functions and $\bigotimes \Theta$ is the joint probability distribution. Since $\bigotimes \Theta$ involves all the variables, it will be a huge object to represent explicitly.

What we want to compute is the result of marginalizing out (eliminating) all the variables from $\bigotimes \Theta$. For a standard influence diagram we have both chance and decision variables, and we eliminate chance variables with a sum operator, and decision variables with a max operator. We can compute the maximum expected utility by performing a sequence of sum and max eliminations to $\bigotimes \Theta$. Performing combinations leads to functions involving larger sets of variables, which is expensive in terms of both computational cost and time. One therefore would like to delay performing computations where possible. Thus, when eliminating a variable X, with, for example, the $\sum$ operator, one transforms Θ to a collection Θ', including only functions that don't involve X, such that $\sum_X (\bigotimes \Theta) = \bigotimes \Theta'$. Crucially, the functions in Θ that don't involve X are left unchanged, so still appear in Θ'. However, these variable elimination computations are not trivial when generalized forms of probability and utility are considered in our proposed framework.

We begin with a standard example of influence diagrams, and show in detail how a variable elimination algorithm computes expected utility (Section 2). We will extend this algorithm for a wide range of formalisms. More precisely, we consider generalized forms of probability values and utility values, and associated generalized uncertainty and utility functions, and consider what properties are needed for the algorithm to be correct (Section 3). We describe some formalisms that satisfy the axioms, including interval-valued utility, multi-objective utility and order of magnitude probability and utility (Section 3.2).

We then show how both chance variables and decision variables are eliminated (Section 4). To eliminate a variable involves replacing the current collection of generalized probability and utility functions with a new set whose combination is equivalent to the marginal of the initial set. We continue by defining generalized influence diagram systems (Section 5) and proving that one can iteratively eliminate all the variables to obtain the maximum value of expected utility for the case where utility values are totally ordered (Section 6).

We go on to consider the case where values of utility are only partially ordered (Section 7); then there will typically not be a unique maximal value of utility, but a set of them. To compute this set we need to perform operations on sets of utility values. We therefore show how to compute, by sequential variable elimination, a set of utility values that is equivalent, in a natural sense, to the set of maximal values of expected utility (Section 8). This therefore allows influence diagram computation for *any* formalisms satisfying the axioms.

Finally, we take a closer look at the case when only rough (or imprecise) estimates of the decision model's parameters are available. We describe in detail the order of magnitude influence diagram which can be viewed as a qualitative theory for influence diagrams in which such partially specified sequential decision problems can be modeled and solved (Section 9). More specifically, the model involves an order of magnitude representation of the probabilities and utilities, and thus allows the decision maker to specify partially ordered preferences via finite sets of utility values. To compute the set of maximal expected utility values we show how to use a variable elimination algorithm that performs efficient operations on sets of utility values involving at most two elements.

1.2. Outline of the Article

Following preliminaries and notations (Section 2), Section 3 introduces the notions of generalized uncertainty and utility values while Section 4 shows how to eliminate chance and decision variables from these structures. In Section 5 we define generalized influence diagram systems and show in Section 6 how to compute the maximum value of expected utility. Section 7 and Section 8 present the technical results for the case of partially ordered utility values. Section 9 is dedicated to the order of magnitude influence diagram formalism, Section 10 overviews related work, while Section 11 provides concluding remarks and directions of future work.

The Appendix contains the proofs and auxiliary material required for the main results. The paper extends earlier work of the authors which was published in [1, 2].

2. Preliminaries

We first define some notation that we will use throughout the paper, and then go through a standard influence diagram example to illustrate the computational of expected utility using iterative variable elimination. In later sections we extend this solution method for more general forms of influence diagram.

2.1. Some Notation

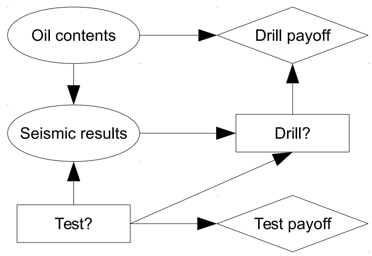
We denote variables by upper case letters (e.g., $X, Y, \ldots$) and values of variables by lower case letters (e.g., $x, y, \ldots$). Sets of variables are denoted by bold upper case letters (e.g., $\mathbf{X} = \{X_1, \ldots, X_n\}$). We denote the set of possible values (also called the *domain*) of variable X_i by $\Omega(X_i)$ or Ω_{X_i}. The expression $X_i = x_i$ (or simply x_i when the variable is clear) denotes an assignment of a value $x_i \in \Omega(X_i)$ to X_i while $\mathbf{X} = \mathbf{x}$ (or simply $\mathbf{x}$) denotes an assignment of values to all variables in $\mathbf{X}$, namely $\mathbf{x} = (X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)$. $\Omega(\mathbf{X})$ (or $\Omega_{\mathbf{X}}$) denotes the Cartesian product of the domains of all variables in $\mathbf{X}$, namely $\Omega(\mathbf{X}) = \Omega(X_1) \times \Omega(X_2) \times \ldots \times \Omega(X_n)$.

The notation $\sum_{\mathbf{x} \in \Omega(\mathbf{X})}$ denotes the sum over all possible configurations of variables in $\mathbf{X}$, namely $\sum_{\mathbf{x} \in \Omega(\mathbf{X})} = \sum_{x_1 \in \Omega(X_1)} \sum_{x_2 \in \Omega(X_2)} \cdots \sum_{x_n \in \Omega(X_n)}$. For brevity, we will abuse notation and write $\sum_{x_i \in \Omega(X_i)}$ (resp. $\max_{x_i \in \Omega(X_i)}$) as $\sum_{X_i}$ (resp. $\max_{X_i}$) and $\sum_{\mathbf{x} \in \Omega(\mathbf{X})}$ (resp. $\max_{\mathbf{x} \in \Omega(\mathbf{X})}$) as $\sum_{\mathbf{X}}$ (resp. $\max_{\mathbf{X}}$).

Given a real valued function f defined over a set of variables $\mathbf{S}$, namely $f : \Omega(\mathbf{S}) \to \mathbb{R}$, and a variable $X \in \mathbf{S}$, the function $(\sum_X f)$ is defined over $\mathbf{U} = \mathbf{S} \setminus \{X\}$ as follows. For every $\mathbf{u} \in \Omega(\mathbf{U})$, $(\sum_X f)(\mathbf{u}) = \sum_{x \in \Omega(X)} f(\mathbf{u}, x)$. Function $(\max_X f)$ is defined in a similar manner: for every $\mathbf{u} \in \Omega(\mathbf{U})$, $(\max_X f)(\mathbf{u}) = \max_{x \in \Omega(X)} f(\mathbf{u}, x)$.

2.2. The Oil Wildcatter Decision Problem

We begin by introducing a simple decision problem which will be used throughout the paper. Consider an oil wildcatter that must decide either to *drill* or *not to drill* for oil at a specific site (we adapted this example from [3]). Before drilling, they could perform a *seismic test* that will help determine the *geological structure* of the site and therefore give an indication of the oil contents underground. The test results can show a *closed* reflection pattern (indication of significant oil), an *open* pattern (indication of some oil), or a *diffuse* pattern (almost no hope of oil). The special value *notest* indicates that the test results will not be available if the seismic test is not done. Figure 1 shows the influence diagram of this decision problem (we defer the formal introduction of influence diagrams until Section 5).



Oil contents	P(O)		
	dry	wet	soak
	0.5	0.3	0.2

Test payoff	U_1(T)
Test?	
yes	-10
no	0

Seismic results		P(S | O,T)			
Oil cnt.	Test?	closed	open	diffuse	notest
dry	yes	0.01	0.03	0.96	0
dry	no	0	0	0	1
wet	yes	0.03	0.94	0.03	0
wet	no	0	0	0	1
soak	yes	0.95	0.04	0.01	0
soak	no	0	0	0	1

Drill payoff		U_2(O,D)
Oil cnt.	Drill?	
dry	yes	-70
dry	no	0
wet	yes	50
wet	no	0
soak	yes	200
soak	no	0

Figure 1: The oil wildcatter influence diagram.

An influence diagram [4, 5] is typically represented by a directed acyclic graph containing nodes for chance variables (depicted as oval-shaped nodes) and decision variables (depicted as rectangle-shaped nodes) as well as the utility functions (depicted as diamond-shaped nodes). For each chance and decision node there is an arc directed from each of its parent variables to it, and there is an arc directed from each variable in the scope of a utility function toward its utility node.

There are therefore two decision variables, T (Test) and D (Drill), and two chance variables S (Seismic results) and O (Oil contents). The probabilistic knowledge consists of the conditional probability distributions $P(O)$ and $P(S|O,T)$, while the utility function is the sum of $U_1(T)$ and $U_2(D,O)$. Therefore, the task is to determine the optimal policies for T and D, denoted by π_T and π_D (details are deferred to Section 5), that maximize the expected utility. Namely,

$$\mathcal{E} = \max_{\pi_T,\pi_D} \sum_{S,O} P(O) \cdot P(S|O,T) \cdot (U_1(T) + U_2(O,D)). \tag{1}$$

The choice of D can depend on its parents S and T, i.e., the decision to drill (D) is made after the decision to test (T), and can depend the the result S of the test. In contrast, the decision to test T has to be made first, without knowledge of the values of any of the other variables. The unobservable chance variable (O) must be placed last in the order. Based on these considerations (see also Section 5 for details), we obtain:

$$\mathcal{E} = \max_T \sum_S \max_D \sum_O P(O) \cdot P(S|O,T) \cdot (U_1(T) + U_2(O,D)). \tag{2}$$

We eliminate variables from right to left, starting with summing out O:

Summing over O:

$$\mathcal{E} = \max_T \sum_S \max_D \phi_O(S,T) \cdot (U_1(T) + \psi_O(S,T,D)) \tag{3}$$

where

$$\phi_O(S,T) = \sum_O P(O) \cdot P(S|O,T) \tag{4}$$

and

$$\psi_O(S,T,D) = \frac{1}{\phi_O(S,T)} \sum_O P(O) \cdot P(S|O,T) \cdot U_2(O,D). \tag{5}$$

Therefore, eliminating the chance variable O (by summation) creates a new probabilistic component ($\phi_O(S,T)$) and a new utility component ($\psi_O(S,T,D)$), respectively. For illustration, Figure 2 shows the detailed tables of $\phi_O(S,T)$ (left) and $\psi_O(S,T,D)$ (right) based on the input functions from Figure 1. For example, $\psi_O(S = \text{closed}, T = \text{yes}, D = \text{yes})$ is equal to $(1/0.204) \times (0.5 \times 0.01 \times -70 + 0.3 \times 0.03 \times 50 + 0.2 \times 0.95 \times 200) = 186.8$.

Maximizing over D:

Pushing expressions not involving D, and eliminating D from Eq. 3 we get:

$$\begin{aligned} \mathcal{E} &= \max_T \sum_S \phi_O(S,T) \cdot (U_1(T) + \max_D \psi_O(S,T,D)) \\ &= \max_T \sum_S \phi_O(S,T) \cdot (U_1(T) + \psi_D(S,T)) \end{aligned} \tag{6}$$

where

Seismic results	$\phi_O(S,T)$			
Test?	closed	open	diffuse	notest
yes	0.204	0.305	0.491	0
no	0	0	0	1

Seismic results		$\psi_O(S,T,D)$			
Test?	Drill?	closed	open	diffuse	notest
yes	yes	186.8	48.0	-66.7	0
yes	no	0	0	0	0
no	yes	0	0	0	20
no	no	0	0	0	0

Figure 2: Intermediate probabilistic component $\phi_O(S,T)$ and utility component $\psi_O(S,T,D)$, respectively.

$$\psi_D(S,T) = \max_D \psi_O(S,T,D). \tag{7}$$

Here, we only generated a utility component ($\psi_D(S,T)$) by maximizing over the relevant sum of utility functions. Notice that $U_1(T)$ does not involve D and therefore it need not be included in the maximization.

Summing over S:

Next, we eliminate S by summation from Eq. 6 and obtain a probabilistic component and a utility component, as follows:

$$\begin{aligned} \mathcal{E} &= \max_T \left(\sum_S \phi_O(S,T) \right) \cdot \left(U_1(T) + \frac{\sum_S \phi_O(S,T) \cdot \psi_D(S,T)}{\sum_S \phi_O(S,T)} \right) \\ &= \max_T \phi_S(T) \cdot (U_1(T) + \psi_S(T)) \end{aligned} \tag{8}$$

where

$$\phi_S(T) = \sum_S \phi_O(S,T) \tag{9}$$

and

$$\psi_S(T) = \frac{1}{\phi_S(T)} \sum_S \phi_O(S,T) \cdot \psi_D(S,T). \tag{10}$$

Figure 3 shows the tables of the intermediate functions generated when eliminating variables D and S, namely $\psi_D(S,T)$, $\phi_S(T)$ and $\psi_S(T)$, respectively.

Seismic results	$\psi_D(S,T)$			
Test?	closed	open	diffuse	notest
yes	186.8	48.0	0	0
no	0	0	0	20

	$\phi_S(T)$
Test?	
yes	1
no	1

	$\psi_S(T)$
Test?	
yes	52.75
no	20.0

Figure 3: Intermediate probabilistic component $\phi_S(T)$ and utility components $\psi_D(S,T)$ and $\psi_S(T)$.

Maximizing over T:

Finally, we find T that maximizes:

$$\mathcal{E} = \max_T \phi_S(T) \cdot (U_1(T) + \psi_S(T)). \tag{11}$$

Plugging in the probability and utility values given in Figure 1 and performing the intermediate numerical calculations (e.g., as shown in Figures 2 and 3), we get: $\phi_S(T = yes) = 1$, $\phi_S(T = no) = 1$, $\psi_S(T = yes) = 52.75$, $\psi_S(T = no) = 20$, $U_1(T = yes) = -10$ and $U_1(T = no) = 0$, respectively. Therefore, solving Eq. 11 yields 42.75 which is the value of the maximum expected utility. The sequence of solution policies is given by the argmax function in Eq. 11 (for T) and Eq. 6 (for D), while respecting the temporal order of the decisions. Specifically, once decision T is made, the value of S will be observed, and then decision D can be made based on T and the observed S. In this case, the optimal policy is to perform the seismic test and to drill only if the test results show an open or closed pattern.

3. Generalized Uncertainty and Utility Values

In standard influence diagrams probability potentials take non-negative real values, and utility functions take real values. In this section, we introduce the notions of generalized probability and utility values along with combination and marginalization operators. Most of the properties of positive reals are still assumed for generalized probability values, the most important exception being that we do not assume a cancellation property for addition (i.e., we do not assume that $a + b = a + c$ only if $b = c$; for instance, the order of magnitude systems in Sections 3.2.3 and 3.2.4 do not satisfy this property). The properties assumed for generalized utility values are much weaker than those satisfied by the real numbers; in particular, we allow partially ordered utility values, which will be useful for expressing imprecise information about utilities, or multi-objective utilities.

We also outline the properties of generalized probability and utility values which will allow correct variable elimination algorithms.

In Section 3.1, we define Uncertainty-Utility Values Structures, the algebraic structures used for generalized probability and utility values, giving the assumptions that are used to ensure the correctness of the variable elimination algorithm for computing generalized expected utility for an influence diagram. (Because we are focusing on computational issues, the properties we assume are much stronger than those assumed in Chu and Halpern's *GEU (Generalized Expected Utility)* system [6].) Section 3.2 gives a number of example formalisms: a form of upper and lower utility (Section 3.2.1), multi-objective utility (Section 3.2.2), and an order of magnitude calculus based on a more qualitative version of probability (Section 3.2.3), which leads naturally to only partially ordered utility values (as do upper and lower utility and multi-objective utility). We also consider (Section 3.2.4), a pair of simpler order of magnitude systems that have totally ordered generalized utility values. In Section 3.3, we define generalized probability and utility functions, and the combination and marginalization operators.

3.1. Uncertainty-Utility Values Structures

An uncertainty values structure (Definition 1) is used to define the values taken by the generalized probability function, along with the corresponding operations on those values; similarly, a utility values structure (Definition 2) defines values and operations for a generalized utility function; these are combined into an uncertainty-utility values structure (Definitions 3 and 4) that also include an operation for combining an uncertainty value and a utility value. In addition, we need a generalized max operator (Definition 5).

Definition 1 (uncertainty values structure)**.** *A* uncertainty values structure *is defined to be a tuple $\langle Q, +, \times, 0, 1\rangle$ that is a positive commutative semi-ring with multiplicative inverses, i.e.,*

- $+$ *and* $\times$ *are both commutative and associative binary operations on set* Q*;*
- $\forall q \in Q$*,* $q + 0 = q$ *(additive identity element* 0*);*
- $\forall q \in Q$*,* $q \times 1 = q$ *(multiplicative identity element* 1*);*
- $\forall q \in Q$*,* $q \times 0 = 0$*, and* $\forall p, q \in Q$*,* $p + q = 0$ *if and only if* $p = q = 0$ *(positive);*
- $\times$ *distributes over* $+$*, namely* $\forall p, q, r \in Q$*,* $(p+q) \times r = (p \times r) + (q \times r)$*;*

- *multiplicative inverses exist for all non-zero elements of Q, so that for all $q \in Q \setminus \{0\}$ there exists some (unique) element $q^{-1} \in Q$ with $q \times q^{-1} = 1$.*

The multiplication operation $\times$ is used for the factored representations of generalized probability functions; the addition operation $+$ is used in the marginalization (summation) operator. The multiplicative inverses are used to allow division as in Equation 5; the positivity condition allows correct treatment of zero denominators in the elimination of a chance variable (Section 4.2). For the standard case, we would use the tuple $\langle \mathbb{R}^+, +, \times, 0, 1 \rangle$, where $\mathbb{R}^+$ is the set of non-negative real numbers.

Definition 2 (utility values structure)**.** *A* utility values structure *is defined to be a tuple $\langle U, +, 0 \rangle$ such that $+$ is a commutative and associative binary operation on set U with identity element 0.*

The addition operation $+$ is used for the factored representations of generalized utility functions. In the standard case, we would use the tuple $\langle \mathbb{R}, +, 0 \rangle$.

An uncertainty-utility values structure combines an uncertainty values structure, representing the generalized probability values, with a utility values structure, representing the generalized utility values. We also need to add another operation $\times_{QU}$ that combines a value of probability with one of utility, in the computation of expected utility. $(*1)$, $(*2)$, $(*3)$ and $(*4)$ are properties used in the variable elimination algorithm. For partially ordered utility cases, it turns out to be useful to consider systems without the last property, leading to what we call a weak uncertainty-utility values structure.

Definition 3 (weak u.u.v. structure)**.** *A* weak uncertainty-utility values structure *(or* weak u.u.v. structure*) is defined to be a tuple $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times_{QU} \rangle$, where $\langle Q, +_Q, \times_Q, 0_Q, 1 \rangle$ is an uncertainty values structure, $\langle U, +_U, 0_U \rangle$ is a utility values structure, and $\times_{QU}$ is a function from $Q \times U \to U$ satisfying the properties $(*1)$, $(*2)$ and $(*3)$ below, for arbitrary $u, u_1, u_2 \in U$ and $q, q_1, q_2 \in Q$ (dropping the $(\cdot)_Q$ and $(\cdot)_U$ subscripts since there is no ambiguity):*

$(*1)$ $1 \times u = u$ *and* $0 \times u = 0_U$*;*

$(*2)$ $q_1 \times (q_2 \times u) = (q_1 \times q_2) \times u$*;*

$(*3)$ $q \times (u_1 + u_2) = (q \times u_1) + (q \times u_2)$.

Definition 4 (u.u.v. structure)**.** *An* uncertainty-utility values structure *(u.u.v. structure) is a weak uncertainty-utility values structure $\mathfrak{U}$ that also satisfies:*

(∗4) $(q_1 + q_2) \times u = (q_1 \times u) + (q_2 \times u)$.

Q will contain the probability-like values, and U will contain the utility-like values. We will usually abbreviate 1_Q to 1, abbreviate $\times_Q$ and $\times_{QU}$ both to $\times$, and $+_Q$ and $+_U$ both to $+$ (the context will make it clear which operation is meant). For the standard case of probability and utility functions, we use uncertainty-utility values structure $\langle \mathbb{R}^+, +, \times, 0, 1, \mathbb{R}, +, 0, \times \rangle$.

When we are eliminating decision variables in an influence diagram computation, we use a $\max$ operator. We generalize this to a disjunctive operation, as defined below. For the totally ordered case (see Section 6), we use $\max$ as the disjunctive operator; for the partially ordered case in Section 7, we use $\max$ over subsets of utility values.

Definition 5 (disjunctive operation)**.** *Let* $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times_{QU} \rangle$ *be a weak u.u.v. structure. Let* $\vee$ *be a binary operation on* U*. We say that* $\vee$ *is a* disjunctive operation *for* $\mathfrak{U}$ *if* $\vee$ *is a commutative and associative operation on* U *such that both* $+_U$ *and* $\times_{QU}$ *distribute over* $\vee$*, so that for any* $q \in Q$ *and all* $u_1, u_2, u_3 \in U$*,* $u_1 + (u_2 \vee u_3) = (u_1 + u_2) \vee (u_1 + u_3)$*, and* $q \times (u_1 \vee u_2) = (q \times u_1) \vee (q \times u_2)$*.*

In all the formalisms we consider, generalized utility values are ordered; our algorithmic approach requires that the operations respect the ordering, in the following sense.

Definition 6 (operation respecting ordering)**.** *Let* T *be some set and let* $\odot$ *be some function from* $T \times U$ *to* U*. We say that* $\odot$ respects *the binary relation* $\succeq$ *on* U *if for all* $t \in T$ *and all elements* u_1, u_2 *of* U*,* $u_1 \succeq u_2 \Rightarrow t \odot u_1 \succeq t \odot u_2$[1]*. We say that weak u.u.v. structure* $\mathfrak{U}$ *respects* $\succeq$ *if both* $+_U$ *and* $\times_{QU}$ *respect* $\succeq$*.*

If $\mathfrak{U}$ respects the total order $\succeq$ and if we define $\vee$ to be maximum with respect to $\succeq$, then it is easy to show that $\vee$ is a disjunctive operation.

Lemma 1. *Let* $\mathfrak{U}$ *be a weak uncertainty-utility values structure that respects total order* $\succeq$*. If we define* $\vee$ *to be maximum with respect to* $\succeq$*, then* $\vee$ *is a disjunctive operation.*

Proof: Write $\mathfrak{U}$ as $\langle Q, +_Q, \times_Q, 0_Q, 1_Q, U, +_U, 0_U, \times_{QU} \rangle$. Operation $\vee$ is clearly a commutative and associative operation. Consider any $q \in Q$ and any $u_1, u_2, u_3 \in$

[1]This can be also be viewed as $\odot$ being monotonic with respect to $\succeq$.

U. We need to show that $u_3+(u_1 \vee u_2) = (u_3+u_1) \vee (u_3+u_2)$, and $q \times (u_1 \vee u_2) = (q \times u_1) \vee (q \times u_2)$.

Since $\succeq$ is a total order, either $u_1 \succeq u_2$ or $u_2 \succeq u_1$. Let us assume that $u_1 \succeq u_2$. (The case when $u_2 \succeq u_1$ follows similarly.) We have $u_1 \vee u_2 = u_1$. Since $+$ and $\times$ respect $\succeq$, we have $u_3 + u_1 \succeq u_3 + u_2$, and $q \times u_1 \succeq q \times u_2$. We thus have $(u_3 + u_1) \vee (u_3 + u_2) = u_3 + u_1 = u_3 + (u_1 \vee u_2)$, and $(q \times u_1) \vee (q \times u_2) = q \times u_1 = q \times (u_1 \vee u_2)$. $\square$

3.2. Example Formalisms

We give next some examples of formalisms that satisfy the axioms defined in Section 3.1 (in particular, Definitions 3 and 4). The results given later in this paper (see Sections 6 and 8) imply that, for any of these formalisms, the maximal values of expected utility can be computed (up to equivalence) by a variable elimination algorithm.

3.2.1. Upper and Lower Utility

It can sometimes be hard to determine precisely a utility value. To allow a representation of imprecise utility, we can let utility functions assign pairs of utility values instead of single values, representing a lower and an upper value of utility. For an influence diagram computation (see Section 5), each policy π, dynamically assigning the decision variables, has an associated lower expected utility LEU(π) and an upper expected utility UEU(π).

A natural ordering[2] on these (expected) utility pairs is the point-wise one given by $\langle u, v\rangle \succeq \langle u', v'\rangle$ if and only if $u \geq u'$ and $v \geq v'$. That is, π is at least as good as π' if LEU(π) $\geq$ LEU(π') and UEU(π) $\geq$ UEU(π'). We then define U to consist of the set of all pairs $\langle u, v\rangle$ of real numbers with $u \leq v$. The sum of two pairs $\langle u_1, v_1\rangle$ and $\langle u_2, v_2\rangle$ is performed point-wise, i.e., to be $\langle u_1 + u_2, v_1 + v_2\rangle$. The additive identity utility element is the pair $\langle 0, 0\rangle$. For probability value p and utility pair $\langle u, v\rangle$, $p \times_{QU} \langle u, v\rangle$ is defined to be $\langle p \times u, p \times v\rangle$. This gives rise to an u.u.v. structure $\langle \mathbb{R}^+, +, \times, 0, 1, \mathbb{R} \times \mathbb{R}, +_U, \langle 0, 0\rangle, \times_{QU}\rangle$ that respects the partial order $\succeq$. Since utility values are only partially ordered, we may well have more than one maximal value of expected utility; we are interested in being able

[2] A more cautious ordering $\succeq'$ is given by $\langle u, v\rangle \succeq' \langle u', v'\rangle$ if and only if $u \geq v'$. However, the addition operation on utility pairs does not respect $\succeq'$, which means that our algorithmic approach cannot then be applied. For example, we have $\langle 3, 5\rangle \succeq' \langle 1, 2\rangle$, because $3 \geq 2$, but $\langle 3, 5\rangle + \langle 0, 2\rangle = \langle 3, 7\rangle \not\succeq' \langle 1, 4\rangle = \langle 1, 2\rangle + \langle 0, 2\rangle$.

to compute the set of maximal (i.e., undominated) values of $\big\langle \mathrm{LEU}(\pi), \mathrm{UEU}(\pi) \big\rangle$ over all policies π.

3.2.2. Multi-Objective Utility

A related system (which is mathematically a generalization) is based on multi-objective utility. One may have more than one independent scale of utility, for example, one based on monetary gain, and one based on risk to health. Again, scalar multiplication and addition are performed point-wise, giving, for m objectives the following u.u.v. structure: $\langle \mathbb{R}^+, +, \times, 0, 1, \mathbb{R}^m, +_U, \langle 0 \times \cdot \times 0 \rangle, \times_{QU} \rangle$, where $+_U$ is the usual, point-wise, addition on $\mathbb{R}^p$, and $\times_{QU}$ is scalar (point-wise) multiplication. We can use the (point-wise) product (Pareto) ordering on $U = \mathbb{R}^p$, given by $\vec{u} \geq \vec{v}$ if and only if for $i = 1, \ldots, m$, $\vec{u}(i) \geq \vec{v}(i)$. Alternatively, since this ordering is a rather weak one, instead one may want to consider imprecise trade-offs between the scales of utility. For example, in a two-objective situation, the decision maker tell us that are happy to gain 3 units of the first objective at the cost of losing one unit of the second, and hence prefer (3, -1) to (0, 0). Such tradeoffs may be elicited using some structured method, or in a more ad hoc way. Therefore, the ordering can be strengthened to take such trade-offs into account, whilst still maintaining the monotonicity properties. The tradeoffs generate a convex cone $\mathcal{C}$, and we define ordering $\succeq$ by $\vec{u} \succeq \vec{v}$ if and only if $\vec{u} - \vec{v}$ is in $\mathcal{C}$ (see also [7, 8] for additional details).

A further related system is multi-agent probability and utilities. Each of a number m of agents makes a judgment of the probability and utility values, which are each represented as vectors of m real values.

3.2.3. Order of Magnitude Calculus

We consider the order of magnitude probability and utility system from [9], which can be viewed as a decision theory for kappa (ranking) functions [10].

Let $\mathcal{O} = \{\langle \sigma, n \rangle \,:\, n \in \mathbb{Z}, \sigma \in \{+, -, \pm\}\,\} \cup \{\langle \pm, \infty \rangle\}$, where $\mathbb{Z}$ is the set of integers. The element $\langle \pm, \infty \rangle$ will sometimes be written as 0, element $\langle +, 0 \rangle$ as 1, and element $\langle -, 0 \rangle$ as -1, respectively. We also define $\mathcal{O}_\pm = \{\langle \pm, n \rangle \,:\, n \in \mathbb{Z} \cup \{\infty\}\,\}$, and $\mathcal{O}_+ = \{\langle +, n \rangle \,:\, n \in \mathbb{Z}\}$.

Elements of $\mathcal{O}$ are interpreted in terms of polynomials (or rational functions) in a parameter ε. ε can be considered as an infinitesimal, or, alternatively, a very small unknown number. $\langle +, n \rangle$ represents a function which is positive and of order ε^n, and $\langle -, n \rangle$ is negative and of order ε^n. When we add a positive and a negative value, both of order ε^n, the answer can be positive or negative, and of order ε^m for any $m \geq n$. We write this imprecise value as $\langle \pm, n \rangle$.

Standard arithmetic operations such as multiplication ($\times$) and addition ($+$) follow from the semantics of the order of magnitude values [9] and are defined next.

Definition 7 (multiplication)**.** *Let $a, b \in \mathcal{O}$ be such that $a = \langle \sigma, m \rangle$ and $b = \langle \tau, n \rangle$. We define $a \times b = \langle \sigma \otimes \tau, m + n \rangle$, where $\infty + n = n + \infty = \infty$ for $n \in \mathbb{Z} \cup \{\infty\}$ and $\otimes$ is the natural multiplication of signs: it is the commutative operation on $\{+, -, \pm\}$ such that $+ \otimes - = -$, $+ \otimes + = - \otimes - = +$, and $\forall \sigma \in \{+, -, \pm\}$, $\sigma \otimes \pm = \pm$.*

This multiplication is associative and commutative, and $\forall a \in \mathcal{O}$, $a \times 0 = 0$ and $a \times 1 = a$. Furthermore, for $b \in \mathcal{O} \setminus \mathcal{O}_{\pm}$, we define b^{-1} to be the multiplicative inverse of b, namely $\langle \sigma, m \rangle^{-1} = \langle \sigma, -m \rangle$ for $\sigma \in \{+, -\}$. Given $a \in \mathcal{O}$, we define $a/b = a \times b^{-1}$.

Definition 8 (addition)**.** *Let $a, b \in \mathcal{O}$ be such that $a = \langle \sigma, m \rangle$ and $b = \langle \tau, n \rangle$. We define $a + b$ to be: (1) $\langle \sigma, m \rangle$ if $m < n$; (2) $\langle \tau, n \rangle$ if $m > n$; (3) $\langle \sigma \oplus \tau, m \rangle$ if $m = n$, where $+ \oplus + = +$, $- \oplus - = -$, and otherwise, $\sigma \oplus \tau = \pm$.*

Addition is associative and commutative, and $a + 0 = a$, $\forall a \in \mathcal{O}$. For $a, b \in \mathcal{O}$, let $-b = -1 \times b$ and $a - b = a + (-b)$. We write $-\langle \sigma, m \rangle = \langle -\sigma, m \rangle$, where $-(+) = -$, $-(-) = +$ and $-(\pm) = \pm$. We also have distributivity: $\forall a, b, c \in \mathcal{O}$, $(a + b) \times c = a \times c + b \times c$.

We use a slightly stronger ordering than that defined in [9] (so that the ordering is respected by the operations).

Definition 9 (ordering)**.** *Let $a, b \in \mathcal{O}$ be such that $a = \langle \sigma, m \rangle$ and $b = \langle \tau, n \rangle$. We define the binary relation $\succ$ on $\mathcal{O}$ by $a \succ b$ if and only if either:*

- *$\sigma = +$ and $\tau = +$ and $m < n$; or*
- *$\sigma = +$ and $\tau = \pm$ and $m \leq n$; or*
- *$\sigma = +$ and $\tau = -$; or*
- *$\sigma = \pm$ and $\tau = -$ and $m \geq n$;*
- *$\sigma = -$ and $\tau = -$ and $m > n$.*

We define relation $\succeq$ by, $a \succeq b \iff a \succ b$ or $a = b$.

In particular, all positive elements are better than all negative elements. We write $a \preceq b$ if and only if $b \succeq a$, and $a \prec b$ if and only if $b \succ a$.

We define $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1_Q, U, +_U, 0_U, \times_{QU} \rangle$ as an u.u.v. structure for the order of magnitude case, where U is $\mathcal{O}$ and Q is $\mathcal{O}_+ \cup \{0\}$, respectively. It can be shown that the previously stated properties including ($*1$), ($*2$), ($*3$), and ($*4$) all hold, and the operations respect the ordering $\succeq$.

Proposition 1. *Define $\mathfrak{U}^{\mathcal{O}}$ to be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}, +, 0, \times \rangle$. Then $\mathfrak{U}^{\mathcal{O}}$ is an uncertainty-utility values structure that respects partial order $\succeq$, i.e., $\succeq$ is respected by $+$ and the operation $\times : \mathcal{O}_+ \cup \{0\} \times \mathcal{O} \to \mathcal{O}$.*

3.2.4. Simplified Order Of Magnitude Calculus

We consider two simplified versions of the Order of Magnitude system, which involve totally ordered utility values: the Simplified Lower OOM (SLOOM) and Simplified Upper OOM (SUOOM). SLOOM and SUOOM are concerned with the subset $\mathcal{O}' = (\mathcal{O} - \mathcal{O}_\pm) \cup \{0\}$ consisting of zero and the positive and negative elements of $\mathcal{O}$.

$\mathcal{O}'$ is totally ordered by $\succeq$ and closed under multiplication $\times$. However, it is not closed under the OOM addition; we define new versions of addition on $\mathcal{O}'$, for SLOOM and SUOOM, as follows.

Definition 10 (SLOOM addition $+_*$ on $\mathcal{O}'$)**.** *We define, for non-zero elements of $\mathcal{O}'$,*

$$(\sigma, m) +_* (\sigma', n) = \begin{cases} (\sigma, m) & \text{if } m < n; \\ (\sigma', n) & \text{if } m > n; \\ (\sigma \boxplus_* \sigma', m) & \text{if } m = n \end{cases}$$

where $+ \boxplus_* + = +$, and otherwise, $\sigma \boxplus_* \sigma' = -$. We also define $a +_* 0 = 0 +_* a = a$ for all $a \in \mathcal{O}'$.

Definition 11 (SUOOM addition $+^*$ on $\mathcal{O}'$)**.** *We define, for non-zero elements of $\mathcal{O}'$,*

$$(\sigma, m) +^* (\sigma', n) = \begin{cases} (\sigma, m) & \text{if } m < n; \\ (\sigma', n) & \text{if } m > n; \\ (\sigma \boxplus^* \sigma', m) & \text{if } m = n \end{cases}$$

where $- \boxplus^* - = -$, and otherwise, $\sigma \boxplus^* \sigma' = +$. We also define $a +^* 0 = 0 +^* a = a$ for all $a \in \mathcal{O}'$.

Operations $+_*$ and $+^*$ are commutative and associative, and distribute over multiplication $\times$. We let $\mathfrak{U}_L^{\mathcal{O}'}$ be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}', +_*, 0, \times \rangle$, and let $\mathfrak{U}_U^{\mathcal{O}'}$ be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}', +^*, 0, \times \rangle$. It can be shown (see Proposition 15 in the appendix) that $\mathfrak{U}_L^{\mathcal{O}'}$ and $\mathfrak{U}_U^{\mathcal{O}'}$ are both uncertainty-utility values structures that respect total order $\succeq$.

3.3. Combining and Marginalizing Uncertainty and Utility Functions

So far, in Sections 3.1 and 3.2, we have only considered the values that generalized probability and utility functions will take. In this section we define the generalized probability and utility functions, and show how the operations on values generate combination and marginalization operations on functions.

We consider a set of (discrete) variables that is partitioned into subsets $\mathbf{X}$ and $\mathbf{D}$, where the elements of $\mathbf{X}$ are known as *chance variables*, and the elements of $\mathbf{D}$ are known as *decision variables*. Let $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times_{QU} \rangle$ be a weak u.u.v. structure. Then:

Definition 12 ($\mathfrak{U}$-uncertainty function)**.** *A* $\mathfrak{U}$-uncertainty function *over variables* $\mathbf{X} \cup \mathbf{D}$ *is a function* $\mathbf{P}$ *from* $\Omega(\mathbf{S})$ *to* Q*, for some* $\mathbf{S} \subseteq \mathbf{X} \cup \mathbf{D}$*, known as the* scope *of* $\mathbf{P}$*, and denoted by* $sc(\mathbf{P})$*.*

Definition 13 ($\mathfrak{U}$-utility function)**.** *A* $\mathfrak{U}$-utility function *over variables* $\mathbf{X} \cup \mathbf{D}$ *is a function* $\mathbf{U}$ *from* $\Omega(\mathbf{S})$ *to* U*, for some* $\mathbf{S} \subseteq \mathbf{X} \cup \mathbf{D}$*, where* $\mathbf{S}$ *is the* scope $sc(\mathbf{U})$ *of* $\mathbf{U}$*.*

Thus, a $\mathfrak{U}$-uncertainty function over $\mathbf{X} \cup \mathbf{D}$ is a generalized probability function over $\mathbf{X} \cup \mathbf{D}$, and a $\mathfrak{U}$-utility function is a generalized utility function.

We say that $\mathbf{P}$ *involves* Y if $sc(\mathbf{P}) \ni Y$. If $\mathbf{T} \supseteq \mathbf{S} = sc(\mathbf{P})$ and $\mathbf{x} \in \Omega(\mathbf{S})$ then we also write $\mathbf{P}(\mathbf{x})$ as an abbreviation for $\mathbf{P}(\mathbf{x}^{\downarrow \mathbf{S}})$, where $\mathbf{x}^{\downarrow \mathbf{S}}$ is the projection of $\mathbf{x}$ to variables $\mathbf{S}$. Similarly, for $\mathfrak{U}$-utility function $\mathbf{U}$.

The operation $\times_Q$ gives rise to a combination operation over generalized probability functions, as in a product of probability functions in a Bayesian network decomposition:

Definition 14 (combination by $\times_Q$)**.** *Let* Φ *be a collection (i.e., a multi-set) of* $\mathfrak{U}$*-uncertainty functions over* $\mathbf{X} \cup \mathbf{D}$*. We define their combination* $\prod \Phi = \prod_{\mathbf{P} \in \Phi} \mathbf{P}$ *to be the* $\mathfrak{U}$*-uncertainty function with scope* $\mathbf{S} = \bigcup_{\mathbf{P} \in \Phi} sc(\mathbf{P})$*, given by, for* $\mathbf{x} \in \Omega(\mathbf{S})$*,* $(\prod \Phi)(\mathbf{x}) = \prod_{\mathbf{P} \in \Phi} (\mathbf{P}(\mathbf{x}))$*, where the last use of* $\prod$ *refers to repeated application of the (associative and commutative) operation* $\times_Q$*.*

The operation $+_U$ generates a combination operation over generalized utility functions, as in an additive decomposition of standard utility functions:

Definition 15 (combination by $+_U$)**.** *Let Ψ be a collection (i.e., multi-set) of $\mathfrak{U}$-utility functions over $\mathbf{X} \cup \mathbf{D}$. We define their combination $\sum \Psi = \sum_{\mathbf{U} \in \Psi} \mathbf{U}$ to be the $\mathfrak{U}$-utility function with scope $\mathbf{S} = \bigcup_{\mathbf{U} \in \Psi} sc(\mathbf{U})$, given by, for $\mathbf{x} \in \Omega(\mathbf{S})$, $(\sum \Psi)(\mathbf{x}) = \sum_{\mathbf{U} \in \Psi} \mathbf{U}(\mathbf{x})$, where the last $\sum$ refers to iterative use of $+_U$.*

The operation $\times_{QU}$ gives rise to an operation for combining a generalized probability function with a generalized utility function, as used in a computation of expected utility:

Definition 16 (combination by $\times_{QU}$)**.** *Let $\mathbf{P}$ be a $\mathfrak{U}$-uncertainty function and let $\mathbf{U}$ be a $\mathfrak{U}$-utility function. We define their combination $\mathbf{P} \times \mathbf{U}$ to be a $\mathfrak{U}$-utility function with scope $\mathbf{S} = sc(\mathbf{P}) \cup sc(\mathbf{U})$ given by $(\mathbf{P} \times \mathbf{U})(\mathbf{x}) = \mathbf{P}(\mathbf{x}) \times_{QU} \mathbf{U}(\mathbf{x})$, for any $\mathbf{x} \in \Omega(\mathbf{S})$.*

Thus $\mathfrak{U}$-utility functions can represent expected utility as well as input utility functions. We define next a marginalization operator from an operation on utility values. In the following, we assume that $\odot$ is some commutative and associative operation on U (for example, $\odot = +$ or a disjunctive operator $\vee$).

Definition 17 (marginalisation)**.** *Let $\mathbf{U}$ be a $\mathfrak{U}$-utility function and let $Y \in \mathbf{X} \cup \mathbf{D}$ be a variable in the scope of $\mathbf{U}$. We define $\bigodot_Y \mathbf{U}$ to be the $\mathfrak{U}$-utility function with scope $\mathbf{S} = sc(\mathbf{U}) - \{Y\}$ given by $(\bigodot_Y \mathbf{U})(\mathbf{x}) = \bigodot_{y \in \Omega(Y)} \mathbf{U}(\mathbf{x}y)$, for $\mathbf{x} \in \Omega(\mathbf{S})$, where $\mathbf{x}y$ is assignment $\mathbf{x}$ extended with $Y = y$. This defines operation $\sum_Y$, based on operation $\odot = +$, and $\bigvee_Y$, based on disjunctive operation $\vee$, so that $(\sum_Y \mathbf{U})(\mathbf{x}) = \sum_{y \in \Omega(Y)} \mathbf{U}(\mathbf{x}y)$, and $(\bigvee_Y \mathbf{U})(\mathbf{x}) = \bigvee_{y \in \Omega(Y)} \mathbf{U}(\mathbf{x}y)$.*

4. Elimination of Variables

In this section we will consider an u.u.v. structure $\mathfrak{U}$, and a pair (Φ, Ψ), where Φ (respectively, Ψ) is a collection of $\mathfrak{U}$-uncertainty (respectively, $\mathfrak{U}$-utility) functions over $\mathbf{X} \cup \mathbf{D}$. We assume that each variable in $\mathbf{X} \cup \mathbf{D}$ is involved in some element of $\Phi \cup \Psi$ (if not, then we can delete any non-involved variables).

A pair (Φ, Ψ) will be considered as a compact representation of the (overall) utility function $\prod_{\mathbf{P} \in \Phi} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi} \mathbf{U}$. We write $\bigotimes(\Phi, \Psi) = \prod_{\mathbf{P} \in \Phi} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi} \mathbf{U}$.

We want to compute a generalized expected utility corresponding to the result of iteratively eliminating (in an appropriate order) all variables from $\bigotimes(\Phi, \Psi)$ (as in the expression in Equation 2 in Section 2.2). We need to do this without explicitly computing $\bigotimes(\Phi, \Psi)$, since the latter is a function on an exponentially large product set.

In Sections 4.1 and 4.2 we will show how to eliminate chance variables and decision variables, respectively. In eliminating a chance variable X (i.e., summing over the values of X) we generate a new pair (Φ', Ψ') that doesn't involve X such that $\sum_X \bigotimes(\Phi, \Psi) = \bigotimes(\Phi', \Psi')$. This involves combining (some) functions that involve X; importantly, functions that don't involve X are left as they are; thus combinations are delayed as long as possible, avoiding producing unnecessarily large functions. Similar remarks apply for the elimination of a decision variable.

As for variable elimination algorithms for Bayesian networks or similar computations, this means that under appropriate conditions on the scopes of the functions, the scopes of the produced functions are never too large.

4.1. Elimination of a Chance Variable

Let $\Phi_{\not\ni X}$ be the multi-set containing the elements of Φ not involving variable X, i.e., with $sc(\mathbf{P}) \not\ni X$, and let $\Phi_{\ni X}$ be the other elements in Φ. Let $\mathbf{P}^+ = \prod_{\mathbf{P} \in \Phi_{\ni X}} \mathbf{P}$ be the combination of elements of Φ involving variable X, and let $\mathbf{P}_\Phi^X$ be $\sum_X \mathbf{P}^+$. Let $\Phi' = \Phi_{\not\ni X} \cup \{\mathbf{P}_\Phi^X\}$.

Similarly, let $\Psi_{\not\ni X}$ be the multi-set containing the elements of Ψ not involving X, and let $\Psi_{\ni X}$ be the other elements in Ψ. For $\mathbf{U} \in \Psi_{\ni X}$, define $\mathbf{U}^{-X}$ to be $\frac{1}{\mathbf{P}_\Phi^X} \times \sum_X (\mathbf{P}^+ \times \mathbf{U})$ (where, in this equation, we define $\frac{1}{0}$ to be 0; this isn't important, but it could sometimes help the efficiency). Hence if $\mathbf{P}_\Phi^X(\mathbf{y}) = 0$ then $\mathbf{U}^{-X}(\mathbf{y}) = 0$. Note that, because of the positivity assumption on the uncertainty values ($p + q = 0 \iff p = q = 0$), $\mathbf{P}_\Phi^X(\mathbf{y}) = 0$ holds if and only if for all $x \in \Omega(X)$, $\mathbf{P}^+(\mathbf{x}y) = 0$. Let $\Psi' = \Psi_{\not\ni X} \cup \{\mathbf{U}^{-X} : \mathbf{U} \in \Psi_{\ni X}\}$.

We define $\sum_X(\Phi, \Psi)$ to be (Φ', Ψ'). Note that no element of Φ' or of Ψ' involves X. In contrast with e.g., the classic method in Jensen *et al* [11], we don't combine together the utility functions involving X when eliminating a chance variable X, since it's not necessary (and combining them can sometimes generate very large functions, making the algorithm much less efficient).

Theorem 1 (eliminating a chance variable)**.** *Let $\mathfrak{U}$ be an uncertainty-utility values structure, with associated summation operation $\sum$, let Φ be a collection of $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$, let Ψ be a collection of $\mathfrak{U}$-utility functions over $\mathbf{X} \cup \mathbf{D}$. Then for any $X \in \mathbf{X}$ which is involved in some element of Φ,*

$$\sum_X \bigotimes(\Phi, \Psi) = \bigotimes\big(\sum_X (\Phi, \Psi)\big).$$

In other words, using the definitions of Φ' and Ψ' given above,

$$\sum_X \Big(\prod_{\mathbf{P} \in \Phi} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi} \mathbf{U} \Big) = \prod_{\mathbf{P} \in \Phi'} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi'} \mathbf{U}.$$

Note that the theorem expresses a combination followed by marginalization as a kind of marginalization followed by combination, with the latter typically being more efficient to directly compute.

To illustrate this result, consider the example in Section 2.2, and let $\Phi = \{P(O), P(S|O,T)\}$, and $\Psi = \{U_1(T), U_2(O,D)\}$. Then, $\bigotimes(\Phi, \Psi) = P(O) \cdot P(S|O,T) \cdot (U_1(T) + U_2(O,D))$. We have $\Phi_{\not\ni O} = \emptyset$ and $\mathbf{P}^+ = P(O) \times P(S|O,T)$; also, $\mathbf{P}_\Phi^O = \sum_O \mathbf{P}^+ = \phi_O(S,T)$ (see Equation 4), and $\Phi' = \{\phi_O(S,T)\}$. We have $\Psi_{\not\ni O} = \{U_1(T)\}$ and $\Psi_{\ni O} = \{U_2(O,D)\}$. Letting $\mathbf{U} = U_2(O,D)$, we obtain $\mathbf{U}^{-O} = \frac{1}{\phi_O(S,T)} \times \sum_O (P(O) \times P(S|O,T) \times U_2(O,D))$, which equals $\psi_O(S,T,D)$ (see Equation 5). Thus, $\Psi' = \{U_1(T), \psi_O(S,T,D)\}$. The theorem shows that $\sum_O \bigotimes(\Phi, \Psi) = \bigotimes(\Phi', \Psi')$, which equals $\phi_O(S,T) \cdot (U_1(T) + \psi_O(S,T,D))$ (cf. Equation 3).

We give now several lemmas that will help us prove Theorem 1. We write u.u.v. structure $\mathfrak{U}$ as $\langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times_{QU} \rangle$.

Lemma 2. *Let $q, q_1, \ldots, q_k \in Q$ and $u, u_1, \ldots, u_k \in U$.*

(i) $\sum_{i=1}^k (q \times u_i) = q \times \sum_{i=1}^k u_i$.

(ii) $\sum_{i=1}^k (q_i \times u) = (\sum_{i=1}^k q_i) \times u$.

Proof: Part (i) follows by iterative application of Property $(*3)$, and (ii) follows by iterative application of Property $(*4)$. □

Lemma 3. *Let $\mathbf{P}$, $\mathbf{P}_1$ and $\mathbf{P}_2$ be $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$ and let $\mathbf{U}$, $\mathbf{U}_1$ and $\mathbf{U}_2$ be $\mathfrak{U}$-utility functions over $\mathbf{X} \cup \mathbf{D}$. Let Φ be a finite (multi-)set of $\mathfrak{U}$-uncertainty functions, and let Ψ be a finite (multi-)set of $\mathfrak{U}$-utility functions. Let X be a variable. The following all hold:*

(i) $(\mathbf{P}_1 \times \mathbf{P}_2) \times \mathbf{U} = \mathbf{P}_1 \times (\mathbf{P}_2 \times \mathbf{U})$*;*

(ii) $\mathbf{P} \times (\mathbf{U}_1 + \mathbf{U}_2) = (\mathbf{P} \times \mathbf{U}_1) + (\mathbf{P} \times \mathbf{U}_2)$*; and, more generally,* $\mathbf{P} \times \sum_{\mathbf{U} \in \Psi} \mathbf{U} = \sum_{\mathbf{U} \in \Psi} (\mathbf{P} \times \mathbf{U})$*;*

(iii) $\sum_X (\mathbf{U}_1 + \mathbf{U}_2) = \sum_X \mathbf{U}_1 + \sum_X \mathbf{U}_2$*; more generally,* $\sum_X \sum_{\mathbf{U} \in \Psi} \mathbf{U} = \sum_{\mathbf{U} \in \Psi} \sum_X \mathbf{U}$*;*

(iv) $\sum_X (\mathbf{P} \times \mathbf{U}) = \big(\sum_X \mathbf{P}\big) \times \mathbf{U}$ *if* $sc(\mathbf{U}) \not\ni X$*, i.e., if* $\mathbf{U}$ *doesn't involve variable* X*;*

(v) $\sum_X (\mathbf{P} \times \mathbf{U}) = \mathbf{P} \times \sum_X \mathbf{U}$ *if* $sc(\mathbf{P}) \not\ni X$*.*

Proof: Part (i) follows using Property $(*2)$. Let $\mathbf{S} = sc(\mathbf{P}_1) \cup sc(\mathbf{P}_2) \cup sc(\mathbf{U})$. $sc((\mathbf{P}_1 \times \mathbf{P}_2) \times \mathbf{U}) = sc(\mathbf{P}_1 \times (\mathbf{P}_2 \times \mathbf{U})) = \mathbf{S}$. For $\mathbf{y} \in \Omega(\mathbf{S})$, $((\mathbf{P}_1 \times \mathbf{P}_2) \times \mathbf{U})(\mathbf{y}) = (\mathbf{P}_1(\mathbf{y}) \times \mathbf{P}_2(\mathbf{y})) \times \mathbf{U}(\mathbf{y})$, which equals, by Property $(*2)$, $\mathbf{P}_1(\mathbf{y}) \times (\mathbf{P}_2(\mathbf{y}) \times \mathbf{U}(\mathbf{y})) = (\mathbf{P}_1 \times (\mathbf{P}_2 \times \mathbf{U}))(\mathbf{y})$, proving the result.

Part (ii) follows similarly, using Lemma 2(i).

Part (iii) follows from associativity and commutativity of addition in U.

(iv): Let $\mathbf{y}$ be an assignment to the scope of $\sum_X (\mathbf{P} \times \mathbf{U})$. Then $\big(\sum_X (\mathbf{P} \times \mathbf{U})\big)(\mathbf{y}) = \sum_{x \in \Omega(X)} \big(\mathbf{P}(\mathbf{y}x) \times \mathbf{U}(\mathbf{y})\big)$, (using $\mathbf{U}(\mathbf{y}x) = \mathbf{U}(\mathbf{y})$, since $\mathbf{U}$ does not involve X). By Lemma 2(ii), this equals $\big(\sum_{x \in \Omega(x)} \mathbf{P}(\mathbf{y}x)\big) \times \mathbf{U}(\mathbf{y})$, which equals $\big(\big(\sum_X \mathbf{P}\big) \times \mathbf{U}\big)(\mathbf{y})$, as required.

(v): This follows in a similar fashion to (iv), but using Lemma 2(i). $\square$

Lemma 4. *Let* $\mathbf{U}^- = \sum_{\mathbf{U} \in \Psi_{\not\ni X}} \mathbf{U}$*, and let* $\mathbf{U}^+ = \sum_{\mathbf{U} \in \Psi_{\ni X}} \mathbf{U}$*. We further define*

$$\mathbf{U}' = \frac{1}{\mathbf{P}_\Phi^X} \times \sum_X (\mathbf{P}^+ \times \mathbf{U}^+).$$

With the notation defined above, we have:

(i) $\mathbf{U}' = \sum_{\mathbf{U} \in \Psi_{\ni X}} \frac{1}{\mathbf{P}_\Phi^X} \times \sum_X (\mathbf{P}^+ \times \mathbf{U})$*;*

(ii) $\mathbf{U}^- + \mathbf{U}' = \sum_{\mathbf{U} \in \Psi'} \mathbf{U}$.

(iii) $\sum_X (\mathbf{P}^+ \times \mathbf{U}^-) = \mathbf{P}_\Phi^X \times \mathbf{U}^-$.

(iv) $\sum_X (\mathbf{P}^+ \times \mathbf{U}^+) = \mathbf{P}_\Phi^X \times \mathbf{U}'$.

Proof: (i): By definition of $\mathbf{U}^+$, $\mathbf{U}'$ equals $\frac{1}{\mathbf{P}_\Phi^X} \times \sum_X (\mathbf{P}^+ \times \sum_{\mathbf{U} \in \Psi_{\ni X}} \mathbf{U})$. Using Lemma 3(ii) and then (iii), this equals $\frac{1}{\mathbf{P}_\Phi^X} \times \sum_{\mathbf{U} \in \Psi_{\ni X}} \sum_X (\mathbf{P}^+ \times \mathbf{U})$. Applying Lemma 3(ii) again gives the result.

(ii): By part (i), $\mathbf{U}'$ equals $\sum_{\mathbf{U} \in \Psi_{\ni X}} \mathbf{U}^{-X}$. Hence, $\sum_{\mathbf{U} \in \Psi'} \mathbf{U} = \sum_{\mathbf{U} \in \Psi_{\not\ni X}} \mathbf{U} + \sum_{\mathbf{U} \in \Psi_{\ni X}} \mathbf{U}^{-X} = \mathbf{U}^- + \mathbf{U}'$.

(iii) follows from Lemma 3(iv).

(iv): If $\mathbf{P}^X_\Phi(\mathbf{y}) = 0$ then, because of the positivity assumption in Definition 1, for all $x \in \Omega_X$, $\mathbf{P}^+(\mathbf{y}x) = 0$, which implies that the functions on both sides of the equation are equal to 0, for such a $\mathbf{y}$. If $\mathbf{P}^X_\Phi(\mathbf{y}) \neq 0$ then the equality follows immediately from the definition of $\mathbf{U}'$. □

We are now ready to give the proof of Theorem 1.

Proof of Theorem 1:
Using Lemmas 3 and 4 we have:

$$\sum_X \left(\prod_{\mathbf{P}\in\Phi} \mathbf{P} \times \sum_{\mathbf{U}\in\Psi} \mathbf{U} \right) =^a \sum_X \Big(\mathbf{P}^- \times \big(\mathbf{P}^+ \times (\mathbf{U}^- + \mathbf{U}^+) \big) \Big)$$

$$=^b \mathbf{P}^- \times \sum_X \big(\mathbf{P}^+ \times (\mathbf{U}^- + \mathbf{U}^+) \big).$$

$$\sum_X \big(\mathbf{P}^+ \times (\mathbf{U}^- + \mathbf{U}^+) \big) =^c \sum_X \big(\mathbf{P}^+ \times \mathbf{U}^- \big) + \sum_X \big(\mathbf{P}^+ \times \mathbf{U}^+ \big)$$

$$=^d \mathbf{P}^X_\Phi \times (\mathbf{U}^- + \mathbf{U}').$$

Putting things together:

$$\sum_X \left(\prod_{\mathbf{P}\in\Phi} \mathbf{P} \times \sum_{\mathbf{U}\in\Psi} \mathbf{U} \right) = \mathbf{P}^- \times \mathbf{P}^X_\Phi \times (\mathbf{U}^- + \mathbf{U}')$$

$$=^e \prod_{\mathbf{P}\in\Phi'} \mathbf{P} \times \sum_{\mathbf{U}\in\Psi'} \mathbf{U}.$$

Equality (a) uses Lemma 3 (i); (b) uses Lemma 3 (v); (c) uses Lemma 3 (ii) and (iii); Equality (d) uses Lemma 4(iii) and (iv), and Lemma 3 (ii); and (e) uses Lemma 4(ii).

4.2. Elimination of a Decision Variable

Here we consider how to eliminate a decision variable D, using a disjunctive operation $\vee$. More specifically, we will define an operation $\bigvee_D$ applied to a pair (Φ, Ψ) (representing a collection generalized uncertainty and utility functions), such that the combination $\bigotimes(\bigvee_D(\Phi, \Psi))$ is equal to the marginalization of the combination of (Φ, Ψ).

We say that $\mathbf{P}$ *does not depend on variable* $Y \in sc(\mathbf{P})$ if for all $y, y' \in \Omega(Y)$ and $\mathbf{x} \in \Omega(\mathbf{S})$ where $\mathbf{S} = sc(\mathbf{P}) - \{Y\}$, $\mathbf{P}(\mathbf{x}y) = \mathbf{P}(\mathbf{x}y')$. If so, we define $\mathbf{P}^{-Y}$ to have scope $\mathbf{S}$ and be given by $\mathbf{P}^{-Y}(\mathbf{x}) = \mathbf{P}(\mathbf{x}y)$ (for any $y \in \Omega(Y)$). Recall that $\Psi_{\ni D}$ is the collection of elements of Ψ involving D, with the other ones forming $\Psi_{\not\ni D}$.

Theorem 2 (elimination of a decision variable)**.** *Let $\vee$ be a disjunctive operation for weak uncertainty-utility values structure $\mathfrak{U}$. Let D be a decision variable in $\mathbf{D}$, let Φ be a collection of $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$, none of which depend on variable D, and let Ψ be a collection of $\mathfrak{U}$-utility functions over $\mathbf{X} \cup \mathbf{D}$. Define $\bigvee_D(\Phi, \Psi) = (\Phi^{-D}, \Psi'')$, where $\Phi^{-D} = \{\mathbf{P}^{-D} : \mathbf{P} \in \Phi\}$, and $\Psi'' = \Psi_{\not\ni D} \cup \{\bigvee_D \sum_{\mathbf{U} \in \Psi_{\ni D}} \mathbf{U}\}$. Then,*

$$\bigvee_D \bigotimes(\Phi, \Psi) = \bigotimes(\bigvee_D(\Phi, \Psi)),$$

i.e.,

$$\bigvee_D \Big(\prod_{\mathbf{P} \in \Phi} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi} \mathbf{U}\Big) = \prod_{\mathbf{P} \in \Phi^{-D}} \mathbf{P} \times \sum_{\mathbf{U} \in \Psi''} \mathbf{U},$$

Considering again the Oil Wildcatter example in Section 2.2, let $\Phi = \{\phi_O(S, T)\}$, and $\Psi = \{U_1(T), \psi_O(S, T, D)\}$ (see Equation 3), so $\Psi_{\not\ni D} = \{U_1(T)\}$, and $\Psi_{\ni D} = \{\psi_O(S, T, D)\}$. We have $\bigotimes(\Phi, \Psi) = \phi_O(S, T) \cdot (U_1(T) + \psi_O(S, T, D))$. The function $\phi_O(S, T)$ does not depend on D since it does not involve D, and $\Phi^{-D} = \Phi$. Then, $\Psi'' = \{U_1(T), \max_D \psi_O(S, T, D)\}$. The theorem implies that $\bigvee_D \bigotimes(\Phi, \Psi)$ equals $\phi_O(S, T) \cdot (U_1(T) + \max_D \psi_O(S, T, D))$ (cf. Equation 6).

To prove Theorem 2, we will use the following lemma:

Lemma 5. *Let D be a decision variable in $\mathbf{D}$, let Φ be a collection of $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$, and let $\mathbf{P}$ be an $\mathfrak{U}$-uncertainty function over $\mathbf{X} \cup \mathbf{D}$. Let $\mathbf{U}_1$ and $\mathbf{U}_2$ be $\mathfrak{U}$-utility functions over $\mathbf{X} \cup \mathbf{D}$.*

(i) If for all $\mathbf{P} \in \Phi$, $\mathbf{P}$ does not depend on D then $(\prod \Phi)$ does not depend on D and $(\prod \Phi)^{-D} = \prod_{\mathbf{P} \in \Phi^{-D}} \mathbf{P}$.

(ii) If $\mathbf{P}$ does not depend on D then $\bigvee_D(\mathbf{P} \times \mathbf{U}) = \mathbf{P}^{-D} \times \bigvee_D \mathbf{U}$.

(iii) If $\mathbf{U}_1$ does not involve D then $\bigvee_D(\mathbf{U}_1 + \mathbf{U}_2) = \mathbf{U}_1 + \bigvee_D \mathbf{U}_2$.

Proof: Part (i) follows easily. The proofs of parts (ii) and (iii) are very similar to the proofs of (v) and (iv), respectively, of Lemma 3, using the two distributivity properties of disjunctive operation $\vee$. $\square$

Proof of Theorem 2: Applying Lemma 5 (i) and (ii) gives:

$$\bigvee_D \Big(\prod_{\mathbf{P}\in\Phi} \mathbf{P} \times \sum_{\mathbf{U}\in\Psi} \mathbf{U} \Big) = \prod_{\mathbf{P}\in\Phi^{-D}} \mathbf{P} \times \bigvee_D \sum \Psi.$$

Applying Lemma 5 (iii) to $\mathbf{U}_1 = \sum \Psi_{\not\ni D}$ and $\mathbf{U}_2 = \sum \Psi_{\ni D}$ gives $\bigvee_D \sum \Psi = \sum \Psi_{\not\ni D} + \bigvee_D \sum \Psi_{\ni D}$, which equals $\sum \Psi''$, completing the proof.

5. Influence Diagram Systems

Theorems 1 and 2 show how to eliminate chance and decision variables, respectively, using marginalization operators $\sum_X$ and $\bigvee_D$ on a pair (Φ, Ψ). We would like to iteratively apply these to eliminate all variables, leading to the maximum expected utility. However, Theorem 2 requires that, before eliminating decision variable D, the current set of uncertainty functions does not depend on D. To ensure that this condition holds for the iterative computation, we require restrictions on the elimination ordering, as well as additional structure on the input collection of uncertainty functions Φ.

Let $\mathbf{P}$ be an $\mathfrak{U}$-uncertainty function with scope $\mathbf{S} \subseteq \mathbf{X} \cup \mathbf{D}$. We say that $\mathbf{P}$ is a *constant function* if it does not depend on any variable, i.e., there exists some value $q \in Q$ such that for all $\mathbf{x} \in \Omega(sc(\mathbf{P}))$, $\mathbf{P}(\mathbf{x}) = q$. We say that $\mathbf{P}$ is a *conditional $\mathfrak{U}$-uncertainty function on X* if $X \in sc(\mathbf{P})$ and $\sum_X \mathbf{P}$ is a constant function. Following previous work [4, 5], we have that:

Definition 18 (ID-system)**.** *An* Influence Diagram system *(ID-system) over $\mathfrak{U}$ is a pair $\langle G, (\Phi, \Psi)\rangle$, such that:*

- *G is a directed acyclic graph on $\mathbf{X} \cup \mathbf{D}$.*
- *$\Phi = \{\mathbf{P}_X : X \in \mathbf{X}\}$, where $\mathbf{P}_X$ is a conditional $\mathfrak{U}$-uncertainty function on X with scope $\{X\} \cup pa_G(X)$; ($pa_G(X)$ means the set of parents of X, i.e., $\{Y : (Y, X) \in G\}$);*
- *each element of Ψ is a $\mathfrak{U}$-utility function (whose scope is a subset of $\mathbf{X} \cup \mathbf{D}$).*

- *G restricted to $\mathbf{D}$ is a total order, which we write as $D_1, \ldots, D_m$.*
- *If $(X, D_i) \in G$ then $(X, D_j) \in G$ for any $j > i$ (this is the* no forgetting *condition).*

Collection Φ is a Bayesian network-style decomposition of a global uncertainty distribution, namely $\prod \Phi$. Also, $\sum \Psi$ represents the overall utility function. Hence the function $\bigotimes(\Phi, \Psi)$ represents the (generalized) probability times utility function.

The pair $(Y, D_i) \in G$ means that the value of Y is known when choosing the value of decision variable D_i. Let $\mathbf{S}_i$ be $pa_G(D_i)$ for $i = 1, \ldots, m$. The choice of value of D_i can therefore depend on the values of variables in $\mathbf{S}_i$ but no others.

Definition 19 (policy)**.** *A* policy *for an ID-system $\langle G, (\Phi, \Psi)\rangle$ over $\mathfrak{U}$ is a sequence $(\pi_1, \ldots, \pi_m)$ where π_i is a function from $\Omega(\mathbf{S}_i)$ to $\Omega(D_i)$; this represents what value of D_i is chosen given the available information: the already observed chance variables, and previous choices of decision variables.*

A policy π determines a value for each decision variable D_i (which depends on the parents set $\mathbf{S}_i$). Given a utility function $\mathbf{U}$ involving all the chance variables $\mathbf{X}$, a policy π determines a utility function $[\mathbf{U}]_\pi$ that involves no decision variables, by assigning their values using π. The expected utility given policy π is given by $EU_\pi = \sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_\pi$. If the utility values set U is totally ordered by relation $\succeq$ then maximum expected utility is $\max_\pi EU_\pi$, where $\max$ is taken with respect to $\succeq$.

For $k = 0, \ldots, m-1$, let $\mathbf{I}_k$ be the set of chance node parents of D_{k+1} that are not parents of any D_i, for $i \leq k$. Hence $\mathbf{I}_k$ is the set of chance nodes that D_{k+1} depends on, but no earlier D_i depends on. We also let $\mathbf{I}_m$ be the other chance nodes, that are not parents of any decision node.

Definition 20 (legal elimination sequence)**.** *A* legal elimination sequence *for an ID-system is a permutation $Y_1, \ldots, Y_n$ of the variables in $\mathbf{X} \cup \mathbf{D}$ that extends the relation $<$ given by: $\mathbf{I}_0 < D_1 < \mathbf{I}_1 < \cdots < D_m < \mathbf{I}_m$, so that, for $i = 0, \ldots, m$, each element X of $\mathbf{I}_i$ comes after D_i (if $i \geq 1$) and before D_{i+1} (if $i < m$).*

For example, if $Y_j \in pa(D_1)$ and $Y_k \in \mathbf{I}_1$ then we must have $j < k$. Note that the sequence is not necessarily compatible with G, so we could have $i < j$ and $(Y_j, Y_i) \in G$, when Y_i and Y_j are both chance variables.

Let τ be a sequence $Y_1, \ldots, Y_n$ of different elements in $\mathbf{X} \cup \mathbf{D}$, and let $\vee$ be a disjunctive operation for $\mathfrak{U}$. We define $\mathbb{M}_\tau^{+,\vee}(\mathbf{U})$ to mean the result of iterative application of the marginalization corresponding to sequence τ, i.e., $\mathbb{M}_{Y_1}(\mathbb{M}_{Y_2}(\cdots(\mathbb{M}_{Y_n}\mathbf{U})\cdots)$ where $\mathbb{M}_Y$ is $\sum_Y$ if Y is a chance variable, and $\mathbb{M}_Y$ is $\bigvee_Y$ if Y is a decision variable. Note that the marginalization operations are applied from right to left, so that the Y_n-marginalization is performed first.

Suppose that $\mathfrak{U}$ respects total order $\succeq$ on U. As mentioned in Section 3.1, if we define $\vee$ to be maximum with respect to $\succeq$, then $\vee$ is a disjunctive operation. Then, for legal elimination sequence τ, $\mathbb{M}_\tau^{+,\vee}\big(\bigotimes(\Phi,\Psi)\big)$ can be shown (e.g., using Proposition 11 below) to be the maximum value of expected utility over all possible policies, i.e., $\max_\pi EU_\pi$.

Example 1. *Recall the influence diagram of the oil wildcatter problem from Figure 1. In this case we have that $\Phi = \{P(O), P(S|O,T)\}$ and $\Psi = \{U_1(T), U_2(D,O)\}$, respectively. There is a unique legal elimination sequence: $\tau = T, S, D, O$ (so that O is eliminated first). The maximum expected utility, which equals $\mathbb{M}_\tau^{+,\vee}\big(\bigotimes(\Phi,\Psi)\big)$, is thus equal to (where $\bigvee$ here means $\max$):*

$$\bigvee_T \sum_S \bigvee_D \sum_O \big(\bigotimes(\Phi,\Psi)\big).$$

6. Elimination of All Variables

In Section 4 we showed how to eliminate both chance variables and decision variables. In Section 6.1 we show how this can be iterated in order to eliminate all variables for an influence diagram system, enabling computation of maximum expected utility. In Section 6.2 we demonstrate how this algorithmic approach can be implemented within a bucket elimination structure, and illustrate this in Section 6.3 with our running example. Section 6.4 considers the complexity of the algorithm.

6.1. Sequential Elimination Result

We show (Theorem 3 below, using notation defined towards the end of the last section) how all variables can be iteratively eliminated for an ID-system. The proof uses iterative application of Theorems 1 and 2, where the main difficulty is in showing that, when eliminating a decision variable D, none of the uncertainty functions $\mathbf{P}$ depend on D (see the conditions of Theorem 2). The conditions assumed on Φ in an ID-system imply this.

Theorem 3. *Let $\mathfrak{U}$ be an uncertainty-utility values structure with operation $+$ on utility values, and let $\vee$ be a disjunctive operation for $\mathfrak{U}$. Let $\mathfrak{I} = \langle G, (\Phi,\Psi)\rangle$ be an ID-system over $\mathfrak{U}$, and let τ be a legal elimination sequence for $\mathfrak{I}$. Then*

$$\mathbb{M}_\tau^{+,\vee}\big(\bigotimes(\Phi,\Psi)\big) = \bigotimes\big(\mathbb{M}_\tau^{+,\vee}(\Phi,\Psi)\big).$$

The right-hand-side $\bigotimes\big(\mathbb{M}^{+,\vee}_{\tau}\big(\Phi,\Psi\big)\big)$ involves iterative local computations, based on sequential elimination of variables. When eliminating (marginalizing out) a variable Y we only deal with functions involving Y: the ones not involving Y are just left as they are. (The operator $\bigotimes$ on the right-hand-side will be easy and often be redundant, since we'll typically just have a single utility value, representing expected utility, when we've eliminated all the variables.)

Let $\mathfrak{I} = \langle G, (\Phi, \Psi)\rangle$ be an ID-system over $\mathfrak{U}$, and let τ be a legal elimination sequence for $\mathfrak{I}$ (see Definition 20). Define, for $k = 1, \ldots, m$, τ_k to be the part of τ starting just after D_k, i.e., if τ is $Y_1, \ldots, Y_n$ and $Y_j = D_k$ then τ_k is the sequence $Y_{k+1}, \ldots, Y_n$. We also define $\tau_0 = \tau$.

A key result in proving Theorem 3 is the following, which shows that, when we need to eliminate a decision variable using Theorem 2, the appropriate conditions apply.

Proposition 2. *Let $\mathfrak{I} = \langle G, (\Phi, \Psi)\rangle$ be an ID-system over $\mathfrak{U}$, and let τ be a legal elimination sequence for $\mathfrak{I}$. Let τ_k be the part of τ starting just after D_k, i.e., if τ is $Y_1, \ldots, Y_n$ and $Y_j = D_k$ then τ_k is the sequence $Y_{k+1}, \ldots, Y_n$. Let (Φ_k, Ψ_k) be $\mathbb{M}^{+,\vee}_{\tau_k}\big(\Phi,\Psi\big)$ (where if τ_k is empty, $\Phi_k = \Phi$ and $\Psi_k = \Psi$). Then, $\mathbb{M}^{+,\vee}_{\tau}\big(\Phi,\Psi\big)$ is well defined, that is, for each $k = m, \ldots, 1$ and each $\mathbf{P} \in \Phi_k$, $\mathbf{P}$ does not depend on D_k.*

Unfortunately, Proposition 2 seems to be not at all trivial to prove, requiring several ancillary results. We relegate the proof of this to the appendix.

Proof of Theorem 3: We iteratively move the $\bigotimes$ operator outwards (leftwards), exchanging it with an operator of the form $\sum_X$ or of the form $\bigvee_D$. The chance variable elimination result (Theorem 1) ensures that we can correctly make this exchange (i.e., without changing the result) for the $\sum_X$ case. For the $\bigvee_D$ case, Proposition 2 and the decision variable elimination result (Theorem 2) implies that the exchange between the $\bigotimes$ operator and the $\bigvee_D$ operator also preserves equality.

Write $\mathbb{M}^{+,\vee}_{\tau}(\cdot)$ as $\mathbb{M}_{Y_1}(\mathbb{M}_{Y_2}(\cdots(\mathbb{M}_{Y_n}(\cdot))\cdots))$. The left-hand-side of the theorem involves a sequence of $n+1$ operators, with $\bigotimes$ being the rightmost (i.e., the one applied first). The right-hand-side involves the same operators, but with $\bigotimes$ being applied at the end, i.e., leftmost. The idea of the proof is to use Theorems 1 and 2 to move the $\bigotimes$ incrementally from right to left.

For i, j with $1 \leq i \leq j \leq n$ let $\mathbb{M}^{+,\vee}_{\tau[i:j]}$ be an abbreviation for the operator:

$$\mathbb{M}_{Y_i}(\mathbb{M}_{Y_{i+1}}(\cdots(\mathbb{M}_{Y_j}(\cdot))\cdots))$$

We also define $\mathbb{M}^{+,\vee}_{\tau[1:0]}$ to be the identity operator (i.e., that makes no change to its operand).

For $i = 0, \ldots, n-1$, let (Φ^i, Ψ^i) be $\mathbb{M}^{+,\vee}_{\tau[i+1:n]}(\Phi, \Psi)$, and so (Φ^0, Ψ^0) equals $\mathbb{M}^{+,\vee}_{\tau[1:n]}(\Phi, \Psi)$, which is equal to $\mathbb{M}^{+,\vee}_{\tau}(\Phi, \Psi)$. We also define $(\Phi^n, \Psi^n) = (\Phi, \Psi)$.

We'll show, for any $i = 1, \ldots, n$, $\mathbb{M}_{Y_i}(\bigotimes(\Phi^i, \Psi^i)) = \bigotimes(\mathbb{M}_{Y_i}(\Phi^i, \Psi^i))$. If Y_i is a chance variable, so that $\mathbb{M}_{Y_i}$ is $\sum_{Y_i}$, then this follows by Theorem 1. Otherwise, Y_i is a decision variable, and $\mathbb{M}_{Y_i}$ is $\bigvee_{Y_i}$. By Proposition 2, (Φ^i, Ψ^i) doesn't depend on variable Y_i, so we can apply Theorem 2 to give $\mathbb{M}_{Y_i}(\bigotimes(\Phi^i, \Psi^i)) = \bigotimes(\mathbb{M}_{Y_i}(\Phi^i, \Psi^i))$ in this case also.

Applying operator $\mathbb{M}^{+,\vee}_{\tau[1:i-1]}$ to both sides of this equation, and using the fact that $\mathbb{M}_{Y_i}(\Phi^i, \Psi^i) = (\Phi^{i-1}, \Psi^{i-1})$, we have, for all $i = 1, \ldots, n$,

$$\mathbb{M}^{+,\vee}_{\tau[1:i]}(\bigotimes(\Phi^i, \Psi^i)) = \mathbb{M}^{+,\vee}_{\tau[1:i-1]}(\bigotimes(\Phi^{i-1}, \Psi^{i-1})).$$

Hence $\mathbb{M}^{+,\vee}_{\tau[1:n]}(\bigotimes(\Phi^n, \Psi^n)) = \bigotimes(\Phi^0, \Psi^0))$, i.e., $\mathbb{M}^{+,\vee}_{\tau}(\bigotimes(\Phi, \Psi)) = \bigotimes(\mathbb{M}^{+,\vee}_{\tau}(\Phi, \Psi))$, proving the result.

6.2. Bucket Elimination for Totally Ordered Utility Values

Suppose that $\succeq$ is a total order on U such that $+_U$ and $\times_{QU}$ both respect $\succeq$, and again define $\vee$ to be maximum with respect to $\succeq$, which is a disjunctive operation (see Lemma 1). As observed above in Section 5, the left-hand-side $\mathbb{M}^{+,\vee}_{\tau}\big(\bigotimes(\Phi, \Psi)\big)$ in Theorem 3 is equal to maximum value of expected utility over all possible policies, i.e., $\max_\pi EU_\pi$. Hence Theorem 3 implies the correctness of an iterative variable elimination algorithm to compute maximum expected utility. This applies to standard probability and utility functions, and also to the simplified order of magnitude systems (Section 3.2.4).

Algorithm 1 describes an iterative variable elimination procedure corresponding with Theorem 3, called BE, which is based on Dechter's *bucket elimination* framework [12]. It takes as input an ID-system $\langle G, (\Phi, \Psi)\rangle$ as well as a legal elimination order τ of the variables $\mathbf{X} \cup \mathbf{D}$ and outputs the maximum expected utility value. BE uses a bucket structure constructed along τ which associates a single variable with each bucket. The bucket of variable Y, denoted by $\mathcal{B}_Y$, has two sets Φ_Y and Ψ_Y which contains all the input probability functions and the utility functions whose highest variable is Y, respectively (see lines 1–4).

The algorithm processes each bucket, top-down from the last to the first, by a variable elimination procedure that computes new probability components (denoted by ϕ) and utility components (denoted by ψ) which are then placed in corresponding lower buckets (lines 5–15). Let Y be the current variable. If $\mathcal{B}_Y$ is a

Algorithm 1: Bucket Elimination (BE) for Totally Ordered Utility

Data: An ID-system $\langle G, (\Phi, \Psi)\rangle$, legal elimination ordering τ of the variables $\mathbf{X} \cup \mathbf{D}$
Result: Maximum expected utility value
`// Partition the sets` Φ `and` Ψ `into buckets`
1 **foreach** *variable* Y *in the reversed order* τ **do**
2 Create bucket $\mathcal{B}_Y$ and its associated sets Φ_Y and Ψ_Y;
3 Let $\Phi_Y = \{\mathbf{P} \in \Phi \,:\, Y \in sc(\mathbf{P})\}$ and $\Psi_Y = \{\mathbf{U} \in \Psi \,:\, Y \in sc(\mathbf{U})\}$;
4 Let $\Phi = \Phi \setminus \Phi_Y$ and $\Psi = \Psi \setminus \Psi_Y$;
`// Process the buckets`
5 **foreach** *variable* Y *in the reversed order* τ **do**
6 **if** Y *is a chance variable* **then**
7 Let $\phi_Y = \sum_Y \prod_{\phi \in \Phi_Y} \phi$;
8 **foreach** $\psi \in \Psi_Y$ **do**
9 Let $\psi_Y = \frac{1}{\phi_Y} \sum_Y (\prod_{\phi \in \Phi_Y} \phi) \times \psi$;
10 **else if** Y *is a decision variable* **then**
11 **foreach** $\phi \in \Phi_Y$ **do**
12 Let $\mathbf{S} = sc(\phi) \setminus \{Y\}$;
13 Compute ϕ_Y as follows: $\forall \mathbf{x} \in \Omega(\mathbf{S}) \,:\, \phi_Y(\mathbf{x}) = \phi(\mathbf{x}y)$ for any value $y \in \Omega(Y)$;
14 Let $\psi_Y = \max_Y \sum_{\psi \in \Psi_Y} \psi$;
15 Add each of the ϕ_Y (resp. ψ_Y) to the set Φ (resp. Ψ) of the highest bucket corresponding to a variable in $sc(\psi_Y)$ (resp. $sc(\psi_Y)$); If Y is the last variable then add ϕ_Y to Φ_0 and ψ_Y to Ψ_0;
16 **return** $(\prod_{\phi \in \Phi_0} \phi) \times (\sum_{\psi \in \Psi_0} \psi)$

chance bucket, then BE generates the corresponding ϕ_Y component by multiplying all probability components in that bucket and eliminating by summation the bucket variable (line 7) (cf. the definition of $\mathbf{P}_\Phi^Y$ in Section 4.1). The ψ_Y components of the bucket are computed as the average utility, normalized by the bucket's compiled ϕ_Y (this corresponds with the definition of $\mathbf{U}^{-Y}$ in Section 4.1). Notice that the utility components need not be combined in a chance bucket as it was previously done in [11] or [12] (however, this requirement was relaxed in [13]). Alternatively, if $\mathcal{B}_Y$ is a decision variable, BE computes a new utility component ψ_Y component by summing all utility components in that bucket and eliminating the bucket variable using the $\bigvee$ operator (compare with Theorem 2). In this case, the probability components in bucket $\mathcal{B}_Y$ are constants when viewed as functions of the bucket's decision variable. Therefore, for each probability component ϕ in bucket $\mathcal{B}_Y$, BE generates a new probability component ϕ_Y by substituting a value

Bucket O: $\Phi_O = \{P(O), P(S|O,T)\}$ $\Psi_O = \{U_2(D,O)\}$

Bucket D: $\Phi_D = \{\}$ $\Psi_D = \{\}$

Bucket S: $\Phi_S = \{\}$ $\Psi_S = \{\}$

Bucket T: $\Phi_T = \{\}$ $\Psi_T = \{U_1(T)\}$

(a) Initial partitioning into buckets

Bucket O: $\Phi_O = \{P(O), P(S|O,T)\}$ $\Psi_O = \{U_2(D,O)\}$

Bucket D: $\Phi_D = \{\}$ $\Psi_D = \{\psi_O(S,T,D)\}$

Bucket S: $\Phi_S = \{\phi_O(S,T)\}$ $\Psi_S = \{\psi_D(S,T)\}$

Bucket T: $\Phi_T = \{\phi_S(T)\}$ $\Psi_T = \{U_1(T), \psi_S(T)\}$

(b) Buckets during execution

S		$\psi_O(S,T,D)$			
T	D	closed	open	diffuse	notest
yes	yes	186.77	48.03	-66.70	0
yes	no	0	0	0	0
no	yes	0	0	0	20.00
no	no	0	0	0	0

S	$\phi_O(S,T)$			
T	closed	open	diffuse	notest
yes	0.204	0.305	0.401	0
no	0	0	0	1.0

S	$\psi_D(S,T)$			
T	closed	open	diffuse	notest
yes	186.77	48.03	0	0
no	0	0	0	20.00

T	$\psi_S(T)$
yes	52.75
no	20.00

$\psi_T() = 42.75$

T	$\phi_S(T)$
yes	1.0
no	1.0

$\phi_T() = 1.0$

(c) Intermediate functions generated

Figure 4: Execution of the Bucket Elimination algorithm.

of variable Y. Finally, the maximum expected utility value is obtained after eliminating the last variable in the ordering, by combining the constant probability and utility components in the sets Φ_0 and Ψ_0, respectively.

6.3. Example Continued

Consider again the influence diagram from Figure 1, where the sets Φ and Ψ are as follows: $\Phi = \{P(O),\, P(S|O,T)\}$ and $\Psi = \{U_1(T),\, U_2(D,O)\}$. Here, $\bigvee$ means $\max$. The legal elimination ordering is $\tau = T, S, D, O$ so that O is eliminated first (as it is unobservable). We want to compute the expected utility, equalling $\bigvee_T \sum_S \bigvee_D \sum_O \big(\bigotimes(\Phi, \Psi)\big)$, which is equal to $\bigotimes\big(\bigvee_T \sum_S \bigvee_D \sum_O(\Phi, \Psi)\big)$, by Theorem 3 (using the notation in Section 4). The initial partitioning into buckets as well as the schematic execution of algorithm BE on this problem is given in Figure 4 and explained below.

Initially, bucket $\mathcal{B}_O$ contains $\Phi_O = \{P(O), P(S|O,T)\}$ and $\Psi_O = \{U_2(O,D)\}$. Since it is a chance bucket, we generate $\phi_O(S,T) = \sum_O P(O) \cdot P(S|O,T)$ and $\psi_O(S,T,D) = \frac{1}{\phi_O(S,T)} \sum_O P(O) \cdot P(S|O,T) \cdot U_2(O,D)$ which are then placed in the lower buckets $\mathcal{B}_S$ and $\mathcal{B}_D$, respectively (see also Figure 4(b)). So far, what has been computed corresponds with $\sum_O(\Phi, \Psi)$, whose first component is the collection of current probability functions, and second component is the collection of current utility functions, i.e., $\sum_O(\Phi, \Psi)$ equals $\big(\{\phi_O(S,T)\},\, \{U_1(T),\, \psi_O(S,T,D)\}\big)$.

The bucket of D is processed next as a decision bucket and we compute $\psi_D(S,T) = \max_D \psi_O(S,T,D)$ which is placed in bucket $\mathcal{B}_S$. Now, we have a representation of $\bigvee_D \sum_O(\Phi,\Psi)$, which equals $\big(\{\phi_O(S,T)\},\ \{U_1(T),\ \psi_D(S,T)\}\big)$. Next, we eliminate the chance variable S and generate $\phi_S(T) = \sum_S \phi_O(S,T)$, and $\psi_S(T) = \frac{1}{\phi_S(T)} \sum_S \phi_O(S,T) \cdot \psi_D(S,T)$, which are both placed in $\mathcal{B}_T$. We have then computed $\sum_S \bigvee_D \sum_O(\Phi,\Psi) = \big(\{\phi_S(T)\},\ \{U_1(T),\ \psi_S(T)\}\big)$. The last bucket to be processed is $\mathcal{B}_T$. Since it is a decision bucket, we generate $\psi_T() = \max_T U_1(T) + \psi_S(T)$ and $\phi_T() = \phi_S(T = yes)$, respectively. The pair $\big(\{\phi_T()\},\ \{\psi_T()\}\big)$ is then equal to $\bigvee_T \sum_S \bigvee_D \sum_O(\Phi,\Psi)$.

Finally, the maximum expected utility value is $\phi_T() \times \psi_T()$, which is equal to $\bigotimes\big(\bigvee_T \sum_S \bigvee_D \sum_O(\Phi,\Psi)\big)$. The intermediate probability and utility functions generated during the execution are shown in Figure 4(c) and correspond to the initial derivations from Section 2.2.

The optimal policy is computed by applying the argmax function to the combination of probability and utility components of the decision buckets $\mathcal{B}_T$ and $\mathcal{B}_D$. Namely, the optimal decision T is given by $\arg\max_T \phi_S(T) \times (U_1(T) + \psi_S(T))$, namely $T = yes$. Once decision T is made, the value s of S can be observed, and then the optimal decision D is given by $\arg\max_D \psi_O(S = s, D, T = yes)$.

6.4. Complexity

As is usually the case with bucket elimination algorithms [14], the time and space complexity of algorithm BE is exponential in the largest scope of a function generated by the algorithm, which is no more than the *induced width* [14] of G' (ordered according to the elimination sequence), where G' is G with an additional clique $\mathbf{S}$ for each input utility function with scope $\mathbf{S}$. In the context of standard influence diagrams one way to reduce the size of the decision policies (and implicitly the computational complexity) is to remove non-requisite arcs from the diagram [15, 13]. Although we believe that these methods are directly applicable to our proposed formalism we defer this in-depth analysis to future work.

7. Sets of Utility Values for Partially Ordered Case

Let $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times \rangle$ be an uncertainty-utility values structure and let $\mathfrak{I} = \langle G, (\Phi, \Psi) \rangle$ be an ID-system over $\mathfrak{U}$. Suppose that the set of utility values U is only partially ordered, by relation $\succeq$. For finite set A of utility values we can consider the set of maximal ones $\max_{\succeq}(A)$, consisting of all $u \in A$ such that there does not exist a different element $v \in A$ with $v \succeq u$. We are interested in policies that generate a maximal value of expected utility, i.e., values of utility in $\max_{\succeq} \{EU_\pi \ : \ \text{policies } \pi\}$. Since there now can be more than one maximal value of expected utility, we need to consider maximization acting on sets of utility values. A neat way of dealing with this is to consider an extended form of utility function, whose range includes sets of utility values (in an analogous way to the approach taken in Section 4.3 of [16] for the much simpler case where there are just utility functions). The operations on utility values then need to be extended to sets of utility values (see Section 7.1). A difficulty is that we obtain then only a weak uncertainty-utility values structure, because Property $(*4)$ (a distributivity property—see Section 3.1) does not necessarily hold. However, the property does hold if we consider convex sets of utility values (see Section 7.2). This leads to a natural definition of equivalence for sets of utility values, such that Property $(*4)$ holds up to equivalence—see Section 7.3. This is the basis for variable elimination for the extended utility functions case (i.e., whose range includes sets of utility values), considered in Section 8.

7.1. Operations on Sets of Utility Values

For u.u.v. structure $\mathfrak{U}$ with set of utility values U, we extend addition and scalar multiplication of utilities to 2^U, the set of subsets of U, in the obvious way.

Definition 21 (addition, multiplication on 2^U). *Let* $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times\rangle$ *be an uncertainty-utility values structure. For any* $A, B \subseteq U$ *and* $q \in Q$*, define:*

- $A + B = \{a +_U b \ : \ a \in A, b \in B\}$*;*
- $q \times A = \{q \times u : u \in A\}$.

We also define $\mathfrak{U}^*$ to be the tuple $\mathfrak{U}^* = \langle Q, +_Q, \times_Q, 0_Q, 1, 2^U, +, \{0_U\}, \times\rangle$ (using the operations on sets as just defined). The following result holds:

Proposition 3. *For any u.u.v. structure* $\mathfrak{U}$*, the associated tuple* $\mathfrak{U}^*$ *is a weak uncertainty-utility values structure.*

Addition and scalar multiplication of sets distribute over union:

Lemma 6. *Let* $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +, 0, \times\rangle$*, be an uncertainty-utility values structure. For any subsets* A, B, C *of* U*,* $(A \cup B) + C = (A + C) \cup (B + C)$*, and for any* $q \in Q$*,* $q \times (A \cup B) = (q \times A) \cup (q \times B)$.

Lemma 6 implies that union is a disjunctive operation for $\mathfrak{U}^*$ (see Definition 5).

Proposition 4. *For any uncertainty-utility values structure* $\mathfrak{U}$*, the union operation* $\cup$ *is a disjunctive operation for* $\mathfrak{U}^*$.

Proof: Let $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1_Q, U, +_U, 0_U, \times_{QU}\rangle$ and $\mathfrak{U}^* = \langle Q, +_Q, \times_Q, 0_Q, 1_Q, 2^U, +, \{0_U\}, \times\rangle$. Union is a commutative and associative operation on 2^U. The two distributive properties of union are given by Lemma 6. $\square$

Unfortunately, Property $(*4)$ (see Section 3.1) does not hold in general for sets, so $\mathfrak{U}^*$ is not necessarily an uncertainty-utility values structure. $(q_1 + q_2) \times A$ is equal, using the property $(*4)$ for $\mathfrak{U}$, to $\{(q_1 \times a) + (q_2 \times a) : a \in A\}$. On the other hand, $(q_1 \times A) + (q_2 \times A)$ equals $\{(q_1 \times a_1) + (q_2 \times a_2) : a_1, a_2 \in A\}$. This implies that $(q_1 + q_2) \times A$ is a subset of $(q_1 \times A) + (q_2 \times A)$. However, they will very often not be equal. To give a simple example with bi-objective utility, let $q_1 = q_2 = 0.5$, and let $A = \{(1, 0), (0, 1)\}$, using the point-wise operations on pairs of real numbers. $(q_1 + q_2) \times A = 1 \times \{(1, 0), (0, 1)\} = \{(1, 0), (0, 1)\}$, whereas $(q_1 \times A) + (q_2 \times A) = \{(0.5, 0), (0, 0.5)\} + \{(0.5, 0), (0, 0.5)\} = \{(1, 0), (0.5, 0.5), (0, 1)\}$.

However, in Section 7.3.3 we will define an equivalence relation, based on convex closure, that ensures that property $(*4)$ holds up to equivalence.

7.2. Convex Sets and Convex Closure

We develop the notion of convex closure for uncertainty-utility values structures, showing that particular properties of convex closure for the standard case of vector spaces over $\mathbb{R}$ hold also for a more general situation based on an uncertainty-value structure. Convex closure is used in the definition of equivalence in Section 7.3.3. Throughout this section we consider an uncertainty-value structure $\mathfrak{U}$ written as $\langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0, \times\rangle$, and, as usual, we sometimes drop the indices when the context makes it clear which operation is being applied.

Definition 22 (convex set)**.** *Let A be a subset of U. A is said to be* convex *if for any $q_1, q_2 \in Q$ with $q_1 + q_2 = 1$, and for any $a, b \in A$, then $(q_1 \times a) + (q_2 \times b) \in A$.*

Definition 23 (convex closure)**.** *The* convex closure $\mathcal{C}(A)$ *of a (finite or infinite) subset A of U is defined to consist of every element of the form $\sum_{i=1}^{k}(q_i \times a_i)$, where k is an arbitrary natural number, each a_i is in A, each q_i is in Q, and where $\sum_{i=1}^{k} q_i = 1$.*

If any $q_i = 0$ in Definition 23 then it can be dropped without changing the result (by Axiom $(*1)$), so we can assume, without loss of generality, that in such a representation each q_i is non-zero.

The following result gives some basic properties of convex sets, used, in particular, for proving Proposition 5 below.

Lemma 7. *For any $A \subseteq U$,*

(i) $\mathcal{C}(A)$ *is a convex set containing* A*.*

(ii) $\mathcal{C}(A)$ *is equal to the intersection of all convex sets containing* A*, and is therefore the unique smallest convex set containing* A*.*

(iii) A *is convex if and only if* $\mathcal{C}(A) = A$*.*

(iv) $\mathcal{C}(\mathcal{C}(A)) = \mathcal{C}(A)$*.*

(v) If $B \subseteq A$ *then* $\mathcal{C}(B) \subseteq \mathcal{C}(A)$*.*

We give some further properties of convex sets that we will use later. For $A, B \subseteq U$ we define $A \oplus B$ to consist of all elements that are the *convex combination* of an element of A and an element of B, i.e., $c \in A \oplus B$ if and only if there exists $a \in A$, $b \in B$, and $p, q \in Q$ with $p + q = 1$ such that $c = (p \times a) + (q \times b)$.

Proposition 5. *Let* A *and* B *be subsets of* U*, and let* q *be an element of* Q*.*

(i) If A *and* B *are convex then* $A + B$ *is convex.*

(ii) $\mathcal{C}(A + B) = \mathcal{C}(A) + \mathcal{C}(B)$*.*

(iii) $\mathcal{C}(\mathcal{C}(A) + B) = \mathcal{C}(A + B)$*.*

(iv) $\mathcal{C}(\mathcal{C}(A) \cup B) = \mathcal{C}(A \cup B)$*.*

(v) $\mathcal{C}(q \times A) = q \times \mathcal{C}(A)$*.*

(vi) $\mathcal{C}(A \cup B) = \mathcal{C}(A) \cup \mathcal{C}(B) \cup (\mathcal{C}(A) \oplus \mathcal{C}(B))$*.*

The following lemma shows that the distributivity property $(*4)$ holds for convex sets A:

Lemma 8. *If* A *is convex then for all* $q_1, q_2 \in Q$*,* $(q_1 + q_2) \times A = (q_1 \times A) + (q_2 \times A)$*.*

Proof: Consider any $u \in (q_1 + q_2) \times A$. Then $u = (q_1 + q_2) \times a$ for some $a \in A$. Using $(*4)$ (for $\mathfrak{U}$), $u = (q_1 \times a) + (q_2 \times a)$ and hence $u \in (q_1 \times A) + (q_2 \times A)$.

Conversely, consider any $u \in (q_1 \times A) + (q_2 \times A)$. Then $u = (q_1 \times a_1) + (q_2 \times a_2)$ for some $a_1, a_2 \in A$. Let $q = q_1 + q_2$. We first consider the case when $q \neq 0_Q$. Let $a = (\frac{q_1}{q} \times a_1) + (\frac{q_2}{q} \times a_2)$. Since A is convex and $\frac{q_1}{q} + \frac{q_2}{q} = 1$, we have $a \in A$. Also, $q \times a = u$ so $u \in (q_1 + q_2) \times A$, as required.

We now consider the case when $q = 0_Q$. The positivity condition of an uncertainty values structure implies that $q_1 = q_2 = 0_Q$, which implies that $u = 0$. So, in particular, $u = 0_Q \times a_1$, and thus $u \in (q_1 + q_2) \times A$. □

7.3. An Equivalence Relation on Sets of Utility Values

In this section we define an equivalence relation on sets of utility values, based on convex closure. First, in Section 7.3.1, we consider some technical issues concerning maximal elements; we go on, in Section 7.3.2, to define a natural ordering on sets of utility values, which we use, in Section 7.3.3, to define the equivalence relation on sets of utility values.

7.3.1. Regarding Maximal Elements

We assume a partial ordering $\succeq$ that is respected by $\mathfrak{U}$, i.e., $\succeq$ is a partial order on U and operations $+_U$ and $\times_{QU}$ respect $\succeq$. If $a \succeq b$ then we say that a *dominates* b. We write $\succ$ for the strict part of $\succeq$, so that $a \succ b$ if and only if $a \succeq b$ and $a \neq b$.

For $A \subseteq U$, define $\max_{\succeq}(A)$, *the maximal elements of* A, to consist of all $a \in A$ such that there does not exist $b \in A$ with $b \succ a$. Hence, $\max_{\succeq}(A)$ is the set of undominated elements of A.

In many cases, every element of a set A is dominated by some maximal element; in particular this holds if A is finite. This can allow a set of utility values to be summarized by its maximal elements. However, this is not universally true, since we may have infinite chains $a_1 \prec a_2 \prec a_3 \prec \cdots$ which have no upper bound in A (consider, for example, the open interval $(0, 1)$ of the real numbers, which has no maximal elements).

Definition 24. *Let A be a subset of U. We say that A satisfies property* MAX *(with respect to $\succeq$) if for all $a \in A$ there exists some $b \in \max_{\succeq}(A)$ with $b \succeq a$.*

Property MAX is strictly weaker than the well-known ascending chain condition (ACC). If A is finite then A satisfies MAX. For $A \subseteq U$ we define subset $\mathcal{R}_{\succeq}(A)$ of A to consist of all elements of A which are not strictly dominated by some maximal element of A, that is:

$$\mathcal{R}_{\succeq}(A) = \{a \in A \, : \, \nexists b \in \text{max}_{\succeq}(A) \quad \text{such that} \quad b \succ a\}.$$

Clearly, we always have $\max_{\succeq}(A) \subseteq \mathcal{R}_{\succeq}(A)$. If A is such that every element of A is dominated by some maximal element of A (in particular, this is the case if A is finite), then $\mathcal{R}_{\succeq}(A) = \max_{\succeq}(A)$.

Lemma 9. *Let A be a subset of U.*

(i) $\max_{\succeq}(A) \subseteq \mathcal{R}_{\succeq}(A)$

(ii) If A satisfies MAX *then* $\max_{\succeq}(A) = \mathcal{R}_{\succeq}(A)$.

We also have the following result.

Lemma 10. *Let A and B be subsets of U. Then* $\max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B) = \max_{\succeq}(A \cup B)$.

7.3.2. Ordering on Sets of Utilities

For $A, B \subseteq U$ we say that $A \succcurlyeq B$ if every element of B is dominated by some element of A (so that A contains as least as large elements as B), i.e., if for all $b \in B$ there exists $a \in A$ with $a \succeq b$. Relation $\succcurlyeq$ on 2^U is reflexive and transitive relation.

Definition 25 (relation $\approx$). *We define equivalence relation $\approx$ by $A \approx B$ if and only if $A \succcurlyeq B$ and $B \succcurlyeq A$.*

If A is such that every element of A is dominated by some maximal element of A (for example, if A is finite), then $A \approx B$ if and only if $\max_{\succeq}(A) = \max_{\succeq}(B)$.

Lemma 11. *Let A and B be subsets of U. Then, $A \approx B$ implies* $\max_{\succeq}(A) = \max_{\succeq}(B)$. *Furthermore, if A satisfies* MAX *then the converse also holds, so we have $A \approx B$ if and only if* $\max_{\succeq}(A) = \max_{\succeq}(B)$.

The following lemma states that scalar multiplication and addition respect the relation $\approx$.

Lemma 12. *Let A, B and C be subsets of U, and let q be an element of Q. Suppose that $A \approx B$. Then*

(i) $q \times A \approx q \times B$;

(ii) $A + C \approx B + C$;

7.3.3. The Equivalence Relation $\equiv$ Between Utility Sets

We will argue that certain different sets of (expected) utility values can reasonably be considered as equivalent. First of all, if A contains elements u and v with $u \succ v$, then we can consider that A and $A - \{v\}$ are equivalent. The second consideration is based on convex closure. For clarity, let's consider the case where the uncertainty values are probability values (but that the utility values may be partially ordered, such as for multi-objective utility or interval-valued utilities). If an agent can generate an expected utility value u with policy π, and utility value u' with policy π' then they may choose an independent auxiliary event E (e.g. based on a random number generator such as rolling a die) with chance p, and choose π

if E holds and π' otherwise. (From the outside it may not even be possible to tell that they are doing this, since we only see the choices they make.) The expected utility is then $pu + (1-p)u'$. More generally, if one can achieve any of a set A of expected utility values, one can generate any element of $\mathcal{C}(\mathcal{A})$ by using the same kind of procedure.

Definition 26 (relation $\equiv$)**.** *We define equivalence relation $\equiv$ on subsets of U by: $A \equiv B$ if and only if $\mathcal{C}(\mathcal{A}) \approx \mathcal{C}(\mathcal{B})$.*

The definition immediately implies that $\equiv$ is an equivalence relation, i.e., it is reflexive, symmetric and transitive, since $\approx$ is an equivalence relation. Two sets of utility values are therefore considered equivalent if, for every convex combination of elements of one, there is a convex combination of elements of the other which is at least as good (with respect to the partial order $\succeq$ on U).

The following result is used in proving Proposition 9 below, which states that Property $(*4)$ holds up to equivalence for the weak u.u.v. structure $\mathfrak{U}^*$ based on extended utility functions.

Proposition 6. *For any subset A of U, $A \equiv \mathcal{R}_{\succeq}(A)$ and $A \equiv \mathcal{C}(A)$. If A satisfies* MAX *(in particular, if A is finite) then $A \equiv \max_{\succeq}(A)$.*

The following result follows immediately from Lemma 11.

Proposition 7. *Let A and B be subsets of U. Then, $A \equiv B$ implies* $\max_{\succeq}(\mathcal{C}(A)) = \max_{\succeq}(\mathcal{C}(B))$. *Furthermore, if $\mathcal{C}(A)$ and $\mathcal{C}(B)$ satisfy property* MAX *then the converse also holds, so we have $A \equiv B$ if and only if* $\max_{\succeq}(\mathcal{C}(A)) = \max_{\succeq}(\mathcal{C}(B))$.

The equivalence relation $\equiv$ is respected by scalar multiplication, addition and union of subsets of utility values:

Proposition 8. *Let A, B and C be subsets of U, and let q be an element of Q. Suppose that $A \equiv B$. Then*

(i) $q \times A \equiv q \times B$;

(ii) $A + C \equiv B + C$;

(iii) $A \cup C \equiv B \cup C$.

As observed above, Property $(*4)$ does not necessarily hold for sets of utility values. However, it does hold for convex sets (see Lemma 8) and a corresponding property based on $\equiv$ holds generally:

Proposition 9. *Consider u.u.v. structure* $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times\rangle$, *respecting partial order* $\succeq$. *Then the associated weak u.u.v. structure* $\mathfrak{U}^*$ *satisfies the following variant of Property* $(*4)$: *for all* $q_1, q_2 \in Q$ *and for all* $A \in 2^U$, $(q_1 + q_2) \times A \equiv (q_1 \times A) + (q_2 \times A)$.

Proof: We need to show that for any $q_1, q_2 \in Q$ and any $A \in 2^U$, $(q_1 + q_2) \times A \equiv (q_1 \times A) + (q_2 \times A)$. Let $B = \mathcal{C}(A)$. Then B is convex, so we have, by Lemma 8, that (I) $(q_1 + q_2) \times B = (q_1 \times B) + (q_2 \times B)$. By Proposition 6, $A \equiv B$, and by Proposition 8, (II) $(q_1 + q_2) \times A \equiv (q_1 + q_2) \times B$. We can also apply Proposition 8 to give (III) $(q_1 \times B) + (q_2 \times B) \equiv (q_1 \times A) + (q_2 \times A)$. Applying (II) then (I) then (III) gives the result, by transitivity of $\equiv$. $\square$

8. Variable Elimination Based on Sets of Utilities

In this section we show how to use variable elimination to compute, up to equivalence, the set of maximal values of expected utility over all policies, for the partially ordered case. Given a u.u.v. structure $\mathfrak{U}$, we make use of extended utility functions (that assign a set of utility values rather than a single utility value), i.e., using the induced weak u.u.v. structure $\mathfrak{U}^*$, which satisfies the properties of a u.u.v. structure up to equivalence (see Section 7, especially, Proposition 9). Theorem 3 in Section 6 can then be used to show correctness of the variable elimination algorithm.

Defining operations $+'$ *and* $\vee$:. Let $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +_U, 0_U, \times\rangle$ be an u.u.v. structure that respects partial order $\succeq$ on U, and let $\mathfrak{U}^*$ be the associated induced weak u.u.v. structure $\langle Q, +_Q, \times_Q, 0_Q, 1, 2^U, +, \{0_U\}, \times\rangle$. As well as the operation $+$ on sets of utility, we also define operation $+'$ on sets of utility values by

$$A +' B = \mathcal{R}_{\succeq}(A + B).$$

(Recall, from Section 7.3.1, that $\mathcal{R}_{\succeq}(A)$ consists of all elements of A that are not strictly dominated by some maximal element of A.) Similarly, define operation $\vee$ on subsets of U by

$$A \vee B = \mathcal{R}_{\succeq}(A \cup B).$$

If A and B are finite then Lemma 9 implies that $A +' B = \max_{\succeq}(A + B)$ and $A \vee B = \max_{\succeq}(A \cup B)$. The following lemma is an immediate consequence of the fact that for any $A \subseteq U$, $\mathcal{R}_{\succeq}(A) \equiv A$ (see Proposition 6).

Lemma 13. *For any subsets* A, B *of* U, $A +' B \equiv A + B$, *and* $A \vee B \equiv A \cup B$.

8.1. Factoring $\mathfrak{U}^*$ by Equivalence $\equiv$

Define $2^U_{\equiv}$ to be the set of all $\equiv$-equivalence classes of 2^U. For $A \in 2^U$, we write $[A]$ to mean the equivalence class containing A. The operations $+$ and $\cup$ on 2^U, and the scalar multiplication $\times$ all respect the equivalence relation $\equiv$, by Proposition 8. Hence they give rise to well-defined operations on $2^U_{\equiv}$ (which we use the same symbols for), and we have, for $A, B \subseteq U$, and $q \in Q$,

$[A] + [B] = [A + B]$;

$[A] \cup [B] = [A \cup B]$; and

$q \times [A] = [q \times A]$.

The key result below follows using Propositions 3, 4 (Section 7.1) and 9 (Section 7.3.3).

Proposition 10. *Let $\mathfrak{U}^*/\!\equiv$ equal $\langle Q, +_Q, \times_Q, 0_Q, 1_Q, 2^U_{\equiv}, +_U, [\{0_U\}], \times\rangle$. Then $\mathfrak{U}^*/\!\equiv$ is an uncertainty-utility values structure, and $\cup$ is a disjunctive operation for $\mathfrak{U}^*/\!\equiv$.*

The result below follows immediately from Lemma 13.

Lemma 14. *For any subsets A, B of U, $[A] +' [B] = [A] + [B]$, and $[A] \vee [B] = [A] \cup [B]$. Hence $+'$ and $+$ are the same operation on $2^U_{\equiv}$, and $\vee$ and $\cup$ are the same operation on $2^U_{\equiv}$.*

A $\mathfrak{U}$-utility function $\mathbf{U}$ can be mapped in the obvious way to a $\mathfrak{U}^*$-utility function $\mathbf{U}^*$ with the same scope, defined by $\mathbf{U}^*(\mathbf{x}) = \{\mathbf{U}(\mathbf{x})\}$. For collection Ψ of $\mathfrak{U}$-utility functions, define collection $\Psi^* = \{\mathbf{U}^* : \mathbf{U} \in \Psi\}$ of $\mathfrak{U}^*$-utility functions.

The following result shows that $u \in \mathbb{M}^{+,\cup}_{\tau}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$ if and only if there exists some policy whose expected utility is u. (The *no forgetting* condition, that the choice of value of a decision variable can depend on all the earlier chance variables, is crucial here.)

Proposition 11. *Let $\mathfrak{I} = \langle G, (\Phi, \Psi)\rangle$ be an $\mathfrak{U}$-ID-system, and let τ be a legal elimination sequence for $\mathfrak{I}$. Then $\mathbb{M}^{+,\cup}_{\tau}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$ is equal to $\{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi} :$ policies $\pi\}$ which is the set of all possible values of expected utility over all policies for $\mathfrak{I}$, i.e., $\{EU_{\pi} :$ policies $\pi\}$.*

The result implies that the value of $\mathbb{M}^{+,\cup}_{\tau}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$ does not depend on the choice of legal elimination sequence τ for $\mathfrak{I}$, i.e., it is the same for all legal choices of τ.

We now give the central result of the paper which shows how to use an iterative variable elimination algorithm to compute a set of utility values that is equivalent (using Proposition 11) to the set of maximal values of expected utility. The idea of the proof is that, because of Propositions 3 and 9, $\mathfrak{U}^*$ is an u.u.v. structure modulo the equivalence relation. Scalar multiplication, addition and union respect equivalence (Proposition 8). Theorem 3 then can be applied for $\mathfrak{U}^*$ modulo equivalence; also union is $\equiv$-equivalent to $\vee$, and addition is $\equiv$-equivalent to $+'$.

Theorem 4. *Let $\mathfrak{U}$ be be an uncertainty-utility values structure (with operation $+$ on utility values) that respects partial order $\succeq$. Let $+'$ and $\vee$ be the induced operations on set of utility values as defined above. Let $\mathfrak{I} = \langle G, (\Phi, \Psi)\rangle$ be an $\mathfrak{U}$-ID-system, and let τ be a legal elimination sequence for $\mathfrak{I}$. Then,*

$$\mathrm{Max}_{g\succcurlyeq}\big(\mathbb{M}_\tau^{+,\cup}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)\big) \equiv \bigotimes\big(\mathbb{M}_\tau^{+',\vee}\big(\Phi, \Psi^*\big)\big).$$

Proof: Probability-utility functions collection (Φ, Ψ^*) over $\mathfrak{U}^*$ maps to a probability utility functions collection over $\mathfrak{U}^*/\equiv$, which we write as $([\Phi], [\Psi^*])$.

Proposition 10 and Theorem 3 imply that

$$\mathbb{M}_\tau^{+,\cup}\big(\bigotimes\big([\Phi], [\Psi^*]\big)\big) = \bigotimes\big(\mathbb{M}_\tau^{+,\cup}\big([\Phi], [\Psi^*]\big)\big).$$

Now, by Lemma 14, $+'$ over $\mathfrak{U}^*/\equiv$ is exactly the same operation as $+$ over $\mathfrak{U}^*/\equiv$, and, similarly, $\cup$ and $\vee$ are the same operation over $\mathfrak{U}^*/\equiv$. So we have:

$$\mathbb{M}_\tau^{+,\cup}\big(\bigotimes\big([\Phi], [\Psi^*]\big)\big) = \bigotimes\big(\mathbb{M}_\tau^{+',\vee}\big([\Phi], [\Psi^*]\big)\big).$$

This implies that

$$\mathbb{M}_\tau^{+,\cup}\big(\bigotimes\big(\Phi, \Psi^*\big)\big) \equiv \bigotimes\big(\mathbb{M}_\tau^{+',\vee}\big(\Phi, \Psi^*\big)\big).$$

The left-hand-side L of this is a finite subset of U, and so $\mathrm{Max}_{\succeq}(L) \equiv L$, by Proposition 6, completing the proof. (Note that the use of $\vee$ in the right-hand-side is always on finite sets so, for these, $A \vee B = \max_{\succeq}(A \cup B)$.) $\square$

Combining Theorem 4 with Proposition 11 we obtain the following result stating that the set $\bigotimes\big(\mathbb{M}_\tau^{+',\vee}\big(\Phi, \Psi^*\big)\big)$ is equivalent to the set of all possible values of expected utility over all policies.

Corollary 1. *With the notation of Theorem 4, $\{EU_\pi \,:\, \textit{policies } \pi\} \equiv \bigotimes\big(\mathbb{M}_\tau^{+',\vee}\big(\Phi, \Psi^*\big)\big)$.*

8.2. Bucket Elimination for Partially Ordered Utility Values

Corollary 1 implies that an iterative variable elimination algorithm generates a set equivalent to the set of optimal (i.e., undominated) values of expected utility. This therefore applies for influence diagrams based on the systems described in Section 3.2, involving multi-objective utility theory, interval-valued utilities, or the order of magnitude system.

For completeness, the bucket elimination procedure for computing the set of optimal (undominated) values of expected utility of an ID-systems with partially ordered utility values is described by Algorithm 2. Here, we make use of the $+'$ and $\bigvee$ operators defined at the beginning of Section 8. Moreover, $\sum'$ and $\prod$ stand for the repeated application of the $+'$ and $\times$ operators, respectively. The time and space complexity of the algorithm is bounded exponentially by the induced width of the graph ordered according to the input legal elimination order, modulo the constant that bounds the size of the undominated sets of expected utility. The latter can however be hard to predict.

9. Order of Magnitude Influence Diagrams

In this section, we describe in detail the Order of Magnitude Influence Diagram (OOM-ID), a more qualitative model that can be used for modeling and solving partially specified sequential decision problems, especially when the quantitative parameters are specified rather poorly and only rough (or imprecise) estimates are available. The model uses an order of magnitude representation of probabilities and utilities, and therefore allows the decision maker to specify partially ordered preferences via sets of utility values. In this case, there will typically not be a unique maximal value of the expected utility, but rather a set of them. To compute this set and also the corresponding decision policy we use the bucket elimination algorithm that performs efficient operations on sets of utility values. Numerical experiments on selected classes of influence diagrams show that as the quantitative information becomes more precise, the qualitative decision process becomes closer to the standard one.

9.1. Generating Small Equivalent Sets of Order of Magnitude Values

Recall from Section 3.2.3 that we can define an uncertainty-utility values structure for the order of magnitude case, namely $\mathfrak{U}^{\mathcal{O}} = \langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}, +, 0, \times \rangle$. Properties $(*1)$, $(*2)$, $(*3)$, and $(*4)$ all hold, and addition and scalar multiplication respect partial order $\succeq$ (from Definition 9).

Algorithm 2: Bucket Elimination (BE) for Partially Ordered Utility

Data: An ID-system $\langle G, (\Phi, \Psi) \rangle$, legal elimination order τ of the variables $\mathbf{X} \cup \mathbf{D}$, partial order $\succeq$
Result: Set of optimal (undominated) values of expected utility
`// Partition the sets Φ and Ψ into buckets`
foreach *variable* Y *in the reversed order* τ **do**
  Create bucket $\mathcal{B}_Y$ and its associated sets Φ_Y and Ψ_Y;
  Let $\Phi_Y = \{\mathbf{P} \in \Phi \ : \ Y \in sc(\mathbf{P})\}$ and $\Psi_Y = \{\mathbf{U} \in \Psi \ : \ Y \in sc(\mathbf{U})\}$;
  Let $\Phi = \Phi \setminus \Phi_Y$ and $\Psi = \Psi \setminus \Psi_Y$;
`// Process the buckets`
foreach *variable* Y *in the reversed order* τ **do**
  if Y *is a chance variable* **then**
    Let $\phi_Y = \sum'_Y \prod_{\phi \in \Phi_Y} \phi$;
    foreach $\psi \in \Psi_Y$ **do**
      Let $\psi_Y = \frac{1}{\phi_Y} \sum'_Y (\prod_{\phi \in \Phi_Y} \phi) \times \psi$;
  else if Y *is a decision variable* **then**
    foreach $\phi \in \Phi_Y$ **do**
      Let $\mathbf{S} = sc(\phi) \setminus \{Y\}$;
      Compute ϕ_Y as follows: $\forall \mathbf{x} \in \Omega(\mathbf{S}) \ : \ \phi_Y(\mathbf{x}) = \phi(\mathbf{x}y)$ for any value $y \in \Omega(Y)$;
    Let $\psi_Y = \bigvee_Y \sum'_{\psi \in \Psi_Y} \psi$;
  Add each of the ϕ_Y (resp. ψ_Y) to the set Φ (resp. Ψ) of the highest bucket corresponding to a variable in $sc(\psi_Y)$ (resp. $sc(\psi_Y)$); If Y is the last variable then add ϕ_Y to Φ_0 and ψ_Y to Ψ_0;
return $(\prod_{\phi \in \Phi_0} \phi) \times (\sum'_{\psi \in \Psi_0} \psi)$

The key result in this section is Proposition 12, that implies that in the (partially ordered) Order of Magnitude computation (OOM), one needs only to work with sets of values which have either one or two elements. The result refers to the equivalence relation $\equiv$ defined in Section 7.3.3, given by $A \equiv B$ if and only if $\mathcal{C}(A) \approx \mathcal{C}(B)$.

Consider two elements $\langle \sigma, m \rangle$ and $\langle \tau, n \rangle$ in $\mathcal{O}$, where we can assume, without loss of generality, that $m \leq n$. It can be shown that any convex combination (see before Proposition 5) of these two elements is of the form $\langle \theta, l \rangle$ where $l \in [m, n]$ and

if $l < n$ then $\theta = \sigma$;

if $l = n$ then $\theta = \sigma \oplus \tau$ or $\theta = \tau$.

This means that convex sets are of a relatively simple form.

We use the following notation: if $a \in \mathcal{O}$ is the element $\langle \sigma, m \rangle$ then we write $\sigma(a) = \sigma$ and $\hat{a} = m$, so that a equals $\langle \sigma(a), \hat{a} \rangle$. To prove Proposition 12 below we use Lemma 15.

Lemma 15. *Let A be any finite subset of $\mathcal{O}$ with $\max_{\succeq}(A) = A$. Then either $|A| = 1$ or there exists some $\sigma \in \{+, -, \pm\}$ such that $\mathcal{C}(A) = \mathcal{C}(\{\langle \pm, m \rangle, \langle \sigma, n \rangle\})$, where $m = \min\{\hat{a} : a \in A\}$, and $n = \max\{\hat{a} : a \in A\}$, and $m < n$.*

Proposition 12. *Let A be any finite subset of $\mathcal{O}$. Then either*

(i) *$A \equiv \{a\}$ for some $a \in \mathcal{O}$, and a is the unique element of $\max_{\succeq}(A)$; or*

(ii) *there exists $\sigma \in \{+, -, \pm\}$ and integers m, n such that $A \equiv \{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$, and $\{\langle \pm, m \rangle, \langle \sigma, n \rangle\} \subseteq \max_{\succeq}(A)$, and $m = \min\{\hat{a} : a \in \max_{\succeq}(A)\}$, and $n = \max\{\hat{a} : a \in \max_{\succeq}(A)\}$, and $m < n$.*

This implies that, when computing with pairs (q, A), in order to perform variable elimination for OOM-based influence diagrams, we can always replace set A by a set A' which has either one or two elements, such that $A' \equiv A$. This affects the complexity of the procedure, which is related to the size of sets A that are used in the computation.

Proof: [of Proposition 12]. Let A be any finite subset of $\mathcal{O}$. Let $B = \max_{\succeq}(A)$. By finiteness of A and Proposition 6, $A \equiv B$. We have $\max_{\succeq}(B) = B$, so, by Lemma 15, either $|B| = 1$, and hence $A \equiv \{a\}$ for some $a \in \mathcal{O}$, or there exists some and $\sigma \in \{+, -, \pm\}$ such that $\mathcal{C}(B) = \mathcal{C}(\{\langle \pm, m \rangle, \langle \sigma, n \rangle\})$, where $m = \min\{\hat{a} : a \in A\}$, and $n = \max\{\hat{a} : a \in A\}$, and $m < n$. If $|B| \neq 1$ then $B \equiv \{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$, by definition of $\equiv$, and hence $A \equiv \{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$. $\square$

9.2. Operations on (Equivalent) Sets of Order of Magnitude Values

We show how to efficiently perform the required operations on sets of order of magnitude values. Because of the arguments in Section 9.1, we can assume that the subsets A of $\mathcal{O}$ are either singleton sets, or are of the form $\{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$, where $m < n$. As well as the inputs of the operations, the output must be of this form. The operations of interest are summation, maximum, and multiplication by an element of $Q = \mathcal{O}_+$.

Scalar Multiplication. Given A of the appropriate form, and $q \in \mathcal{O}_+$ we need to generate a set $\equiv$-equivalent to $q \times A$. In fact, $q \times A$ itself does the job. Write q as $\langle +, l\rangle$. If $A = \{\langle \sigma, m\rangle\}$ then $q \times A$ is just equal to the singleton set $\{\langle \sigma, l+m\rangle\}$. Otherwise, A is of the form $\{\langle \pm, m\rangle, \langle \sigma, n\rangle\}$ where $m < n$. Then $q \times A$ equals $\{\langle \pm, l+m\rangle, \langle \sigma, l+n\rangle\}$, which is of the required form, since $l+m < l+n$.

Summation. Given the sets A_1 and A_2 of required form as before, we want to compute a set A' that is $\equiv$-equivalent to $A_1 + A_2$. We can write A_i as $\{a_i, b_i\}$ where if $a_i \neq b_i$ then $\sigma(a_i) = \pm$ and $\hat{a}_i < \hat{b}_i$. Then, $A_1 + A_2 \equiv \{a, b\}$ where $a = a_1 + a_2$ and $b = b_1 + b_2$.

We can write b more explicitly as $\langle \sigma(b), \hat{b}\rangle$ where $\hat{b} = \min(\hat{b}_1, \hat{b}_2)$, and $\sigma(b) = +$ if and only if all b_i with minimum $\hat{b}_i$ have $\sigma(b_i) = +$; else $\sigma(b) = -$ if all b_i with minimum $\hat{b}_i$ have $\sigma(b_i) = -$; else $\sigma(b) = \pm$. Similarly for a. If $\sigma(a) \neq \pm$ then $\{a, b\}$ reduces to a singleton because $a = b$.

Maximization. Given the sets $A_1, A_2 \subseteq \mathcal{O}$, both of them having the required form, we want to compute a set A' that is $\equiv$-equivalent to $A_1 \cup A_2$, and so is $\equiv$-equivalent to $\max_{\succeq}(A_1 \cup A_2)$.

As before, we can write A_i as $\{a_i, b_i\}$ where if $a_i \neq b_i$ then $\sigma(a_i) = \pm$ and $\hat{a}_i < \hat{b}_i$. We define A' to be $\{a_1 \vee^\circ a_2, b_1 \vee_\circ b_2\}$, where $a_1 \vee^\circ a_2 = \max_{>^\circ}(a_1, a_2)$ and define $A_\circ = \max_{>_\circ}(b_1, b_2)$, where total orders $>^\circ$ and $>_\circ$ are defined below.

Definition 27. *We define relations $>^\circ$ and $>_\circ$ as follows, for arbitrary $\langle \sigma, m\rangle$ and $\langle \tau, n\rangle$ in $\mathcal{O}$.*

$\langle \sigma, m\rangle >^\circ \langle \tau, n\rangle$ if and only if either $\langle \sigma, m\rangle \succ \langle \tau, n\rangle$ or $\sigma = \pm$ and $m < n$.

$\langle \sigma, m\rangle >_\circ \langle \tau, n\rangle$ if and only if either $\langle \sigma, m\rangle \succ \langle \tau, n\rangle$, or $\tau = \pm$ and $m > n$.

For $a, b \in \mathcal{O}$ we also define $\geq^\circ$ and $\geq_\circ$ by $a \geq^\circ b \iff a >^\circ b$ or $a = b$; and $a \geq_\circ b \iff a >_\circ b$ or $a = b$.

9.3. Formal Justification of Operations on Pairs of OOM Values

In this section we formally justify the results in Section 9.2.

Relation $>^\circ$ is the total order extending $\succ$ that places the $\pm$ elements as high in order as possible. In contrast, $>_\circ$ places the $\pm$ elements as far down in the order as possible. The order $>^\circ$ looks like this:

$\cdots >^\circ \langle +, -1\rangle >^\circ \langle \pm, -1\rangle >^\circ \langle +, 0\rangle >^\circ \langle \pm, 0\rangle >^\circ \langle +, 1\rangle >^\circ \cdots >^\circ \langle \pm, \infty\rangle >^\circ \cdots >^\circ \langle -, 1\rangle >^\circ \langle -, 0\rangle >^\circ \langle -, -1\rangle >^\circ \cdots$.

Total order $>_\circ$ looks like this:

$\cdots >_\circ \langle +, -1\rangle >_\circ \langle +, 0\rangle >_\circ \langle +, 1\rangle >_\circ \cdots >_\circ \langle \pm, \infty\rangle >_\circ \cdots >_\circ \langle \pm, 1\rangle >_\circ \langle -, 1\rangle >_\circ \langle \pm, 0\rangle >_\circ \langle -, 0\rangle >_\circ \langle \pm, -1\rangle >_\circ \langle -, -1\rangle >_\circ \cdots$.

Thus, $>^\circ$ and $>_\circ$ order the $\pm$ elements in opposite ways: $>^\circ$ orders the $\pm$ elements similarly to the positive elements, and $>_\circ$ orders them similarly to the negative elements.

Lemma 16. *$>^\circ$ and $>_\circ$ are both (strict) total orders on $\mathcal{O}$ that extend order $\succ$.*

It is useful to consider maximal elements of a set with respect to both $>^\circ$ and $>_\circ$.

Definition 28. *For finite $A \subseteq \mathcal{O}$, define $A^\circ = \max_{>^\circ}(A)$ and define $A_\circ = \max_{>_\circ}(A)$. Define $\rho(A) = \{A^\circ, A_\circ\}$.*

We have the following characterization of A° and $A_\circ$.

Lemma 17. *Consider any finite $A \subseteq \mathcal{O}$. Then A° is the unique element a of $\max_{\succeq}(A)$ with smallest value of $\hat{a}$, and $A_\circ$ is the unique element b of $\max_{\succeq}(A)$ with largest value of $\hat{b}$.*

We have that $\rho(A)$ is equivalent to A (for finite A):

Proposition 13. *For finite $A \subseteq \mathcal{O}$, $A \equiv \rho(A)$.*

Below are some basic properties of the operator $\rho(\cdot)$.

Lemma 18. *Let A be any finite subset of $\mathcal{O}$. Then,*

(i) $(\rho(A))^\circ = A^\circ$ *and* $(\rho(A))_\circ = A_\circ$*; and*

(ii) $\rho(\rho(A)) = \rho(A)$.

The next lemma gives some basic monotonicity properties of $\geq^\circ$ and $\geq_\circ$.

Lemma 19. *For any $a_1, a_2 \in \mathcal{O}$, and $q \in \mathcal{O}_+$,*

(i) if $a_1 \geq^\circ a_2$ *then* $q \times a_1 \geq^\circ q \times a_2$*;*

(ii) if $a_1 \geq_\circ a_2$ *then* $q \times a_1 \geq_\circ q \times a_2$.

The definition and two lemmas below lead to the key Proposition 14, which implies that a variable elimination computation can be replaced by an $\equiv$-equivalent one just using two-element sets.

Definition 29. *Define operations $\vee^\circ$ and $\vee_\circ$ on $\mathcal{O}$ by $a \vee^\circ b = \max_{>^\circ}(\{a,b\})$ and $a \vee_\circ b = \max_{>_\circ}(\{a,b\})$. Thus $a \vee^\circ b = (\{a,b\})^\circ$ and $a \vee_\circ b = (\{a,b\})_\circ$.*

For finite subsets A and B of $\mathcal{O}$, define operation $\boxplus$ by $A \boxplus B = \{A^\circ + B^\circ, A_\circ + B_\circ\}$.

Define operation $\veebar$ by $A \veebar B = \{A^\circ \vee^\circ B^\circ, A_\circ \vee_\circ B_\circ\}$.

Lemma 20. *Let A and B be finite subsets of $\mathcal{O}$.*

(i) $\rho(A \cup B) = A \veebar B = \rho(A) \veebar \rho(B)$.

(ii) $\rho(q \times A) = q \times \rho(A)$.

Lemma 21. $\rho(A) \boxplus \rho(B) = A \boxplus B \equiv \rho(A) + \rho(B)$.

Proposition 14. *For any finite subsets Let A and B be finite subsets of $\mathcal{O}$ and let q be an element of $\mathcal{O}_+$. Define $q \times' A$ to be $q \times \rho(A)$. Then the following hold.*

(i) $A + B \equiv A \boxplus B$.

(ii) $A \cup B \equiv A \veebar B$.

(iii) $q \times A \equiv q \times' A$.

Proof: (i) Lemma 21 gives that $\rho(A) + \rho(B) \equiv A \boxplus B$. By Proposition 13, we also have $A \equiv \rho(A)$ and $B \equiv \rho(B)$. Proposition 8(ii) then leads to $A + B \equiv \rho(A) + \rho(B)$, and thus $A + B \equiv A \boxplus B$.

(ii) Lemma 20 gives that $\rho(A \cup B) = A \veebar B$. Proposition 13 implies that $A \cup B \equiv \rho(A \cup B)$ and thus $A \cup B \equiv A \veebar B$.

(iii) Lemma 20 implies that $\rho(q \times A) = q \times \rho(A) = q \times' A$. Proposition 13 implies that $q \times A \equiv \rho(q \times A)$ and thus $q \times A \equiv q \times' A$. □

Consider an expression built from finite subsets of $\mathcal{O}$ and operations $\cup$, $+$ and $q \times \cdot$ for various $q \in \mathcal{O}_+$. If one replaces each $\cup$ by $\veebar$, and $+$ by $\boxplus$ and each $\times$ by $\times'$ then it can be shown, using Proposition 14 that one gets an expression which is $\equiv$-equivalent. This can be proved by induction on the number of operations. If there are no operations, i.e., just a set A, then it trivially holds. Otherwise, let us assume that it is true if the expression involves $k-1$ or less operations. Consider an expression with k operators, and consider each case of the top operator. E.g., consider if it's $\cup$, so we have $A \cup B$. If we replace the operators in A we get an expression evaluating to some set A', and similarly for B'. The inductive hypothesis implies that $A \equiv A'$ and $B \equiv B'$. We have $A' \cup B' \equiv A' \veebar B'$. Proposition 8 then implies that $A \cup B \equiv A' \cup B'$, and hence $A \cup B \equiv A' \veebar B'$, i.e., the modified expression is equivalent to the original one. The other cases are similar, so the induction goes through.

What this means is that if one performs a computation with each $\cup$ replaced by $\veebar$, and $+$ by $\boxplus$ and each $\times$ by $\times'$ then we will get an equivalent result. The advantage of using the new operations is that the resulting subset of $\mathcal{O}$ produced at each stage of the calculation has at most two elements.

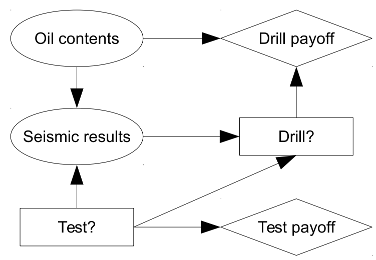
Example 2. *Consider the sets $A = \{\langle\pm, 2\rangle, \langle +, 5\rangle\}$ and $B = \{\langle\pm, 1\rangle, \langle -, 5\rangle\}$. We have $A^\circ = \langle\pm, 2\rangle$, and $A_\circ = \langle +, 5\rangle$, and thus $\rho(A) = A$. Similarly, $\rho(B) = \{B^\circ, B_\circ\} = \{\langle\pm, 1\rangle, \langle -, 5\rangle\} = B$. We also have $\rho(A \cup B) = \{\langle\pm, 1\rangle, \langle +, 5\rangle\}$. It can be seen that $\rho(A \cup B) \equiv A \cup B$, since $\langle +, 5\rangle \succ \langle -, 5\rangle$, and $\langle\pm, 2\rangle$ is a convex combination of $\langle\pm, 1\rangle$ and $\langle +, 5\rangle$, illustrating Proposition 13. The set $A + B$ equals $\{\langle\pm, 1\rangle, \langle\pm, 2\rangle, \langle\pm, 5\rangle\}$. Also, $\rho(A + B) = \{(A + B)^\circ, (A + B)_\circ\} = \{\langle\pm, 1\rangle, \langle\pm, 5\rangle\}$. We further have that $A \boxplus B = \{A^\circ + B^\circ, A_\circ + B_\circ\}$, which equals $\{\langle\pm, 1\rangle, \langle\pm, 5\rangle\} = \rho(A + B)$; and $A \veebar B = \{A^\circ \vee^\circ B^\circ, A_\circ \vee_\circ B_\circ\} = \{\langle\pm, 1\rangle, \langle +, 5\rangle\}$, which equals $\rho(A \cup B)$.*

9.4. The Qualitative Decision Model

An *Order of Magnitude Influence Diagram* (OOM-ID) is an ID-system defined by a pair $\langle G, (\Phi, \Psi)\rangle$, such that (i) G is a directed acyclic graph over $\mathbf{X} \cup \mathbf{D}$; (ii) $\Phi = \{\mathbf{P}_X : X \in \mathbf{X}\}$, where $\mathbf{P}_X$ is a conditional order of magnitude probability distribution on X with scope $\mathbf{S} = \{X\} \cup pa_G(X)$ (where $pa_G(X)$ are the parents of X in G) which maps every configuration of $\mathbf{S}$ to a positive order of magnitude probability value, i.e., $P_X : \Omega(\mathbf{S}) \to \mathcal{O}_+$; (iii) each element ψ of Ψ is an order of magnitude utility function whose scope $\mathbf{S}$ is a subset of $\mathbf{X} \cup \mathbf{D}$, and represents partially ordered preferences of the decision maker, i.e., $\psi : \Omega(\mathbf{S}) \to 2^{\mathcal{O}}$. Collection Φ is a Bayesian network-style factorization of a global order of magnitude probability distribution ($\prod \Phi$), and $\sum \Psi$ represents the overall order of magnitude utility function.

Solving an OOM-ID means finding the set of policies that generate maximal values of order of magnitude expected utility, i.e., values of utility in the set $\max_{\succcurlyeq}\{EU_\pi : \text{policies } \pi\}$. We say that a policy π is *optimal* if the corresponding order of magnitude expected utility EU_π is undominated. The set of all such policies is called *optimal policies set*.

The bucket elimination algorithm for OOM-IDs is obtained from Algorithm 2 by using the summation ($+$), multiplication ($\times$) and maximization ($\max$) operations over partially ordered sets of order of magnitude values introduced in Section 9.2. To obtain an optimal policy, the algorithm processes the decision buckets



Oil contents	P(O)		
	dry	wet	soak
	(+,0)	(+,0)	(+,0)

Test payoff	U_1(T)
Test?	
yes	{(-,-1)}
no	{(+,∞)}

Seismic results		P(S | O,T)			
Oil cnt.	Test?	closed	open	diffuse	notest
dry	yes	(+,2)	(+,1)	(+,0)	(+,∞)
dry	no	(+,∞)	(+,∞)	(+,∞)	(+,0)
wet	yes	(+,1)	(+,0)	(+,1)	(+,∞)
wet	no	(+,∞)	(+,∞)	(+,∞)	(+,0)
soak	yes	(+,0)	(+,1)	(+,2)	(+,∞)
soak	no	(+,∞)	(+,∞)	(+,∞)	(+,0)

Drill payoff		U_2(O,D)
Oil cnt.	Drill?	
dry	yes	{(-,-1)}
dry	no	{(+,∞)}
wet	yes	{(+,-1)}
wet	no	{(+,∞)}
soak	yes	{(+,-2)}
soak	no	{(+,∞)}

Figure 5: The oil wildcatter order of magnitude influence diagram.

in reverse order, from the first decision variable to the last. Each decision rule is generated by taking the argument of the maximization operator applied over the combination of the probability and utility components in the respective bucket, for each configuration of the variables in the bucket's scope (i.e., the union of the scopes of all functions in that bucket minus the bucket variable Y_p). When processing the current decision variable D_k, all of the previous decision variables in the ordering $D_1, \ldots, D_{k-1}$ are set to their already determined optimal values.

9.5. The Oil Wildcatter Decision Problem Revisited

Figure 5 displays the order of magnitude influence diagram corresponding to the oil wildcatter decision problem from Example 1. For our purpose, we used an extension of Spohn's mapping from the original probability distributions and utility functions to their corresponding order of magnitude approximation [17, 18]. Specifically, given a small positive $\epsilon < 1$, the order of magnitude approximation of a probability value $p \in (0, 1]$ is $\langle +, k \rangle$ such that $k \in \mathbb{Z}$ and $\epsilon^{k+1} < p \leq \epsilon^k$, while the order of magnitude approximation of a positive utility value $u > 0$ is $\langle +, -k \rangle$ such that $\epsilon^{-k} \leq u < \epsilon^{-(k+1)}$ (the case of negative utilities is symmetric). For example, if we consider $\epsilon = 0.1$ then the probability $P(S = closed|O = dry, T = yes) = 0.01$ is mapped to $\langle +, 2 \rangle$, while the utilities $U_2(O = dry, D = yes) = -70$ and $U_2(O = soaking, D = yes) = 200$ are

Table 1: Optimal policies sets for the order of magnitude influence diagram from Figure 5.

decision rule		OOM-ID		
		$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$
Test?		{yes,no}	{yes, no}	{yes, no}
Drill?	S=closed, T=yes	yes	yes	{yes, no}
	S=open, T=yes	yes	yes	{yes, no}
	S=diffuse, T=yes	no	{yes, no}	{yes, no}
	S=closed, T=no	yes	yes	{yes, no}
	S=open, T=no	yes	yes	{yes, no}
	S=diffuse, T=no	yes	yes	{yes, no}
$\max_{\succcurlyeq}\{EU_\pi$: policies $\pi\}$		$\{\langle +, -1\rangle\}$	$\{\langle +, 0\rangle\}$	$\{\langle \pm, 0\rangle, \langle +, \infty\rangle\}$

mapped to $\langle -, -1\rangle$ and $\langle +, -2\rangle$, respectively.

Table 1 shows the optimal policies sets (including the maximum order of magnitude expected utility) obtained for the order of magnitude influence diagrams corresponding to $\epsilon \in \{0.1, 0.01, 0.001\}$. When $\epsilon = 0.1$, we can see that there are two optimal policies having the same maximum order of magnitude expected utility, namely π' (for $T = yes$) and π'' (for $T = no$). Therefore, if the seismic test is performed ($T = yes$) then drilling is to be done only if the test results show an open or closed pattern. Otherwise ($T = no$), the wildcatter will drill regardless of the test results. Ties like these at the decision variables are expected given that the order of magnitude probabilities and utilities represent abstractions of the real values. The expected utilities of π' and π'' in the original influence diagram are 42.75 and 20.00, respectively.

When $\epsilon = 0.01$, we also see that both drilling options are equally possible if the seismic test is performed and the test results show a diffuse pattern. In this case, there are four optimal policies having the same maximum order of magnitude expected utility. Finally, when $\epsilon = 0.001$, we can see that all decision options are possible and the corresponding optimal policies set contains 128 policies. The explanation is that the order of magnitude influence diagram contains in this case only trivial order of magnitude values such as $\langle +, 0\rangle$, $\langle -, 0\rangle$ and $\langle +, \infty\rangle$, respectively.

9.6. Numerical Experiments

We evaluate empirically the quality of the decision policies obtained for order of magnitude influence diagrams. These experiments were carried out on a 2.4GHz quad-core processor with 8GB of RAM.

Methodology. We experimented with random influence diagrams described by the parameters $\langle n_c, n_d, k, p, r, a\rangle$, where n_c is the number of chance variables, n_d is the number of decision variables, k is the maximum domain size, p is the number of parents in the graph for each variable, r is the number of root nodes and a is the arity of the utility functions. The structure of the influence diagram is created by randomly picking $n_c + n_d - r$ variables out of $n_c + n_d$ and, for each, selecting p parents from their preceding variables, relative to some ordering, whilst ensuring that the decision variables are connected by a directed path. A single utility node with a parents picked randomly from the chance and decision nodes is then added to the graph.

We generated two classes of random problems with parameters $\langle n, 5, 2, 2, 5, 5\rangle$ and having either positive utilities only or mixed (positive and negative) utilities. They are denoted by $P : \langle n, 5, 2, 2, 5, 5\rangle$ and $M : \langle n, 5, 2, 2, 5, 5\rangle$, respectively. In each case, 75% of the chance nodes were assigned extreme conditional probability distributions (CPDs) which were populated with numbers drawn uniformly at random between 10^{-5} and 10^{-4}, whilst ensuring that the table is normalized (these CPDs are almost deterministic with values very close to 0 or 1). The remaining CPDs were randomly filled using a uniform distribution between 0 and 1. For class P, the utilities are of the form 10^u, where u is an integer uniformly distributed between 0 and 5. For class M, the utilities are of the form $+10^u$ or -10^u, where u is between 0 and 5, as before, and we have an equal number of positive and negative utility values. Each influence diagram instance was then converted into a corresponding order of magnitude influence diagram using the mapping of the probabilities and utilities described in Section 9.5, for some $\epsilon < 1$. Intuitively, the smaller ϵ is, the coarser the order of magnitude approximation of the exact probability and utility values (i.e., more information is lost).

Measures of Performance. To measure how close the decision policies derived from the optimal policy set of an order of magnitude influence diagram are to the optimal policy of the corresponding standard influence diagram, we use two relative errors, defined as follows. Let $\mathcal{I}$ be an influence diagram and let $\mathcal{I}_\epsilon$ be the corresponding order of magnitude approximation, for some ϵ value. We sample s different policies, uniformly at random, from the optimal policies set of $\mathcal{I}_\epsilon$, and for each sampled policy we compute its expected utility in $\mathcal{I}$. Let Δ_{med} be a policy corresponding to the median expected utility v_{med} amongst the samples. We define the relative error $\eta_{med} = |(v - v_{med})/v|$, where v is the maximum expected utility of the optimal policy in $\mathcal{I}$. Similarly, we define $\eta_{max} = |(v - v_{max})/v|$, where Δ_{max} is the best policy having the highest expected utility v_{max} amongst the samples.

Implementation Details. Given an OOM-ID instance $\mathcal{I}_\epsilon$ with m decision variables $D_1, \ldots, D_m$, the bucket elimination algorithm returns an optimal policy $\pi = (\pi_1, \ldots, \pi_m)$ having maximum order of magnitude expected utility. In general, there can be more than one such optimal policies. In our implementation, we actually compute a compact representation of the optimal policies set, as follows. In the second phase of the algorithm we process the decision buckets in reverse order, from first to last (i.e., following the temporal order of the decisions), to generate the decision rules. Let D_p be the current decision bucket. The decision rule π_p is generated by taking the argument of the maximization operator applied over the combination of the probability and utility components in the respective bucket, for each configuration of the variables in the bucket's scope, but we instrument the code to also record the ties (if any). The scope of the bucket is the union of the scopes of all functions in that bucket, minus the bucket variables, and represents all variables before D_p that can influence the decision, namely $pa(D_p)$. Therefore, each entry in the decision rule's table will record all possible options for that particular decision, given the current assignment of the variables in $pa(D_p)$. We call π_p an *augmented* decision rule. Sampling a policy $\pi' = (\pi'_1, \ldots, \pi'_m)$ from the optimal policies set is rather straightforward. The augmented decision rules are processed in the corresponding temporal order (from π_1 to π_m). Let π_i be the current augmented decision rule. We create a decision rule π'_i from π_i by uniformly randomly selecting a decision option for each configuration of the variables in the scope of the decision rule, whilst ensuring that the selected decision option is consistent with all previous decision rules.

Results. Figure 6 displays the distribution of the relative errors η_{med} (top) and η_{max} (bottom) obtained on order of magnitude influence diagrams derived from class P (i.e., positive utilities), as a function of the problem size (given by the number of variables), for $\epsilon \in \{0.5, 0.05, 0.005\}$. Each data point and corresponding error bar represents the 25^{th}, median and 75^{th} percentiles obtained over 30 random problem instances generated for the respective problem size. We can see that η_{med} is the smallest (less than 10%) for $\epsilon = 0.5$. However, as ϵ decreases,

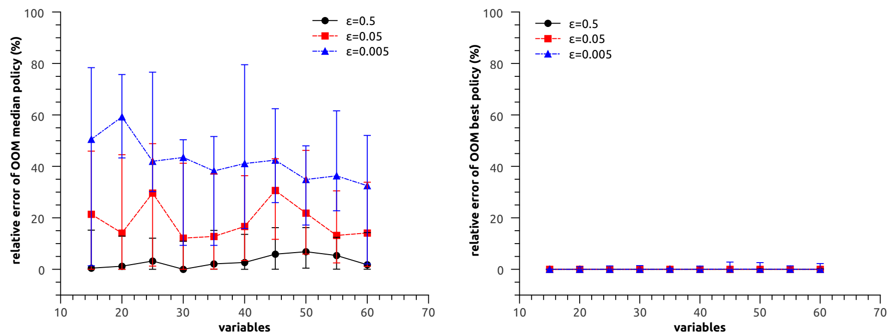


Figure 6: Results for class P influence diagrams. We show the distribution of the relative errors η_{med} (top) and η_{max} (bottom) for $\epsilon \in \{0.5, 0.05, 0.005\}$. # of samples $s = 100$.

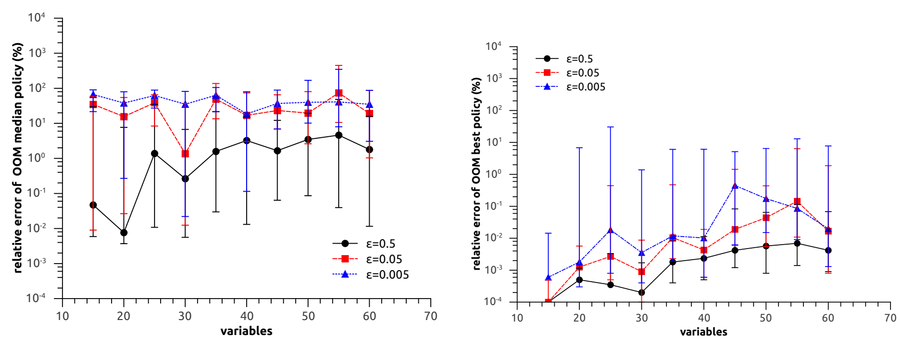


Figure 7: Results for class M influence diagrams. We show the distribution of the relative errors η_{med} (top) and η_{max} (bottom) for $\epsilon \in \{0.5, 0.05, 0.005\}$. # of samples $s = 100$.

the loss of information due to the order of magnitude abstraction increases and the corresponding relative errors η_{med} increase significantly. Notice that the best policy Δ_{max} derived from the order of magnitude influence diagram was almost identical to that of the corresponding standard influence diagram, for all ϵ (i.e., the error η_{max} is virtually zero).

Figure 7 shows the distribution of η_{med} (top) and η_{max} (bottom) obtained on order of magnitude influence diagrams from class M (i.e., mixed utilities). The pattern of the results is similar to that from the previous case. However, in this case, the errors span over two or three orders of magnitude, especially for $\epsilon = 0.05$ and 0.005. This is because the sampled policy space includes policies that are quite different from each other and, although they have the same maximum order of magnitude expected utility, their expected utility in the corresponding standard

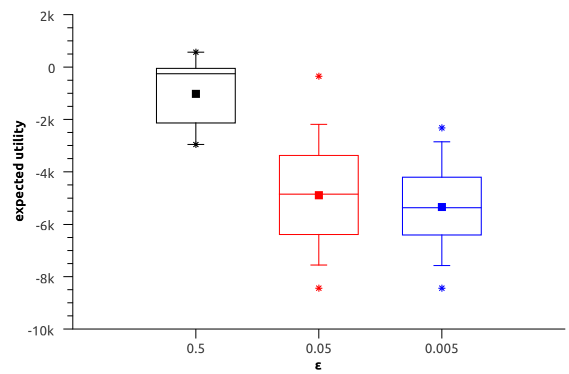


Figure 8: Distribution of the expected utility values for 100 policies sampled uniformly at random for a class $M : \langle 40, 5, 2, 2, 5, 5 \rangle$ order of magnitude influence diagram instance.

influence diagram is significantly different. For this reason, we looked in more detail at the distribution of the expected utility values of policies sampled from the optimal policies set for different values of ϵ.

Figure 8 plots the distribution of the expected utility values of 100 policies sampled uniformly at random from the optimal policies set of a class $M : \langle 40, 5, 2, 2, 5, 5 \rangle$ order of magnitude influence diagram, for $\epsilon \in \{0.5, 0.05, 0.005\}$. The maximum expected utility of the optimal policy in the corresponding class M standard influence diagram is 568.98. As expected, we see that the smallest sample variance is obtained for $\epsilon = 0.5$. In this case, the mean of the samples (i.e., policies) is $-1, 021.39$ and the standard deviation is 1367.29. For $\epsilon = 0.05$ and $\epsilon = 0.005$, the policies are spread out even more from the mean, and the variance of the expected utility is significantly larger. For example, the mean and standard deviation corresponding to $\epsilon = 0.05$ are $-4, 894.94$ and $1, 827.33$, respectively, and for $\epsilon = 0.005$ they are $-5, 338.88$ and $1, 513.58$, respectively. This explains the large variations of the relative errors η_{med} and η_{max} (corresponding to severe degradation in the quality of the decision policies), especially for smaller ϵ values.

Finally, we note that in terms of running time solving the order of magnitude influence diagram instances took almost the same amount of time as the corresponding standard influence diagram instances, thus we omit this comparison.

10. Related Work

The variable elimination approach we use in this paper is based on that for standard influence diagrams, in particular, in Jensen *et al.* [11] and Dechter [12], which builds on previous work such as Shachter and Peot [19], and Shenoy [20]. The work that is closest in spirit to the current work is that by by Pralet, Schiex and Verfaillie [21], who also consider an axiomatic framework for generalized influence diagrams (and other sequential decision making problems), involving a form of generalized expected utility [6]. In contrast with our framework, Pralet's work does not assume division (multiplicative inverses) for the uncertainty values, which allows some more qualitative uncertainty formalisms to be reasoned with. (The existence of multiplicative inverses is used to generate properties we need for convex sets of utility values.) However, Pralet *et al.* focus on the case of totally ordered utility values, with the major contribution of our paper being for the partially ordered case.

Another general computation framework is Valuation Algebras/Networks [22], building on work by Shenoy and Shafer [23]. Since it involves only one marginalization operator it doesn't apply directly for solving influence diagrams, which require a marginalization for eliminating chance nodes, and a different one for eliminating decision nodes. Prakash Shenoy [20, 24] has developed this kind of computational structure for solving influence diagrams based on standard probability and utility. A general axiomatic framework for solving Markov Decision Processes (which have a different and somewhat simpler structure than influence diagrams) is described in [25]; this framework also allows utilities to be only partially ordered. Kikuti and Cozman [26], as well as Kikuti, Cozman and Filho [27] allow interval probabilities (which are not covered by our framework), and focus on precise utility; similarly [28].

Credal networks [29] are a related formalism based on imprecise probability. They generalize classical Bayesian networks by replacing the conditional probability distributions with closed convex sets of probabilities (credal sets) which allow for a richer representation of uncertainty.

Haenni [30] introduced a partially ordered valuation algebra to facilitate approximate inference. The valuation algebra was extended later to compute optimal policies for limited memory influence diagrams [31] and to perform more efficient inference in credal networks [32].

Diehl and Haimes [33] consider influence diagrams with multiple objectives (with just a single multi-objective value node), with the solution methods involving propagation of sets of utility vectors, as in the current paper. López-Díaz et al. [34] consider generalized influence diagrams based on fuzzy random variables.

The work that is closest to our order of magnitude influence diagrams model is that by Bonet and Pearl [35] who consider qualitative MDPs and POMDPs based also on an order of magnitude approximation of probabilities and totally ordered utilities.

Garcia and Sabbadin [36] introduced possibilistic influence diagrams to model and solve decision making problems under qualitative uncertainty in the framework of possibilistic theory.

Multi-objective influence diagrams were developed further in [7] together with an exact variable elimination algorithm as well as an approximation scheme that computes ϵ-coverings of the sets of maximum values of expected utility.

More recently, Cabañas et al [37] proposed an interval-valued quantification of the probability and utility values in influence diagrams. Consequently, they generalized the variable elimination and arc reversal algorithms to cope with these kinds of values. While our formalism and algorithms allow interval utilities it is not yet known if they support interval probabilities as well.

11. Conclusion

In this paper we consider decision making under uncertainty using influence diagrams, but where we allow more general notions of uncertainty than probability and more general notions of utility functions, which, in particular, allow utility values to be only partially ordered. We present an axiomatic framework and list the properties of a formalism that allows maximal (generalized) expected utility to be computed by a sequential variable elimination algorithm. Example formalisms that satisfy the proposed axioms include: decision making under uncertainty based on multi-objective utility, a system of interval-valued utilities, and of multi-agent expected utility, as well as the order of magnitude system. We consider first generalized influence diagram systems where utility values are totally ordered, show how both chance and decision variables can be eliminated effectively, and prove that one can iteratively eliminate all the variables (chance variables by summation, decision variables by maximization) to compute the maximum value of expected utility.

We then consider the case where values of utility are only partially ordered. In this case there will typically not be a unique maximal value of utility, but a set of them, and in order to compute this set we need to perform operations on sets of utility values. We give the central result of the paper (Theorem 4), which shows how to compute by sequential variable elimination a set of utility values that are equivalent to the set of maximal values of expected utility. This ensures soundness (correctness of results) for influence diagram computations for any formalism satisfying the axioms given in Section 3, namely that the formalism forms an uncertainty-utility values (u.-u.v.) structure which respects the partial order on utility values. Thus to show that the variable elimination method applies for a formalism, one just has to verify these axioms (which are simple to check for a given formalism).

We also describe in detail the order of magnitude influence diagram system which involves an order of magnitude representation of the probability and utility values. This is a more qualitative decision model that allows reasoning with imprecise probability and partially ordered imprecise utility values. In order to compute the set of maximal expected utility values (and also the corresponding decision policies) we show how to use a variable elimination algorithm that performs efficient operations on sets of order of magnitude values. More precisely, we prove in Proposition 12 that in the order of magnitude computation one needs only to work with sets of values which have either one or two elements. Therefore, the specialized variable elimination algorithm proposed for solving order of magnitude influence diagrams manipulates sets of order of magnitude values involving at most two elements, which in turn can translate into important time and space savings. Indeed, our empirical evaluation on selected classes of influence diagrams shows that the approach is practical for problems of substantial size. It also demonstrates that as the quantitative information (represented by order of magnitude values) becomes more precise, the qualitative decision process becomes closer to the standard one that involves precise probability and utility values.

Throughout the paper we considered a straight-forward variable elimination algorithm. One might improve this to efficiently make use of constraints (e.g., zero values of the uncertainty and utility functions), building, for instance, on the work of Pralet et al. [21]. An alternative approach is to extend search-based methods, either depth-first or best-first, such as those exploring AND/OR search spaces [38, 39] to solving influence diagrams systems with partially ordered utilities. The main advantage of search methods over variable elimination is that the former can use admissible heuristic evaluation functions to prune unpromising regions of the search space.

[1] R. Marinescu, N. Wilson, Order-of-magnitude influence diagrams, in: Uncertainty in Artificial Intelligence (UAI), 2011, pp. 489–496.

[2] N. Wilson, R. Marinescu, An axiomatic framework for influence diagram computation with partially ordered utilities, in: International Conference on Principles of Knowledge Representation and Reasoning (KR), 2012, pp. 210–220.

[3] H. Raiffa, Decision analysis, Addison-Wesley, 1968.

[4] R. Howard, J. Matheson, Influence diagrams, in: Readings on the Principles and Applications of Decision Analyis, 1984, pp. 721–762.

[5] T. Nielsen, F. Jensen, Byesian Networks and Decision Graphs, Springer-Verlag New York, 2007.

[6] F. Chu, J. Halpern, Great expectations. Part I: On the customizability of Generalized Expected Utility, in: International Joint Conference on Artificial Intelligence (IJCAI), 2003, pp. 291–296.

[7] R. Marinescu, A. Razak, N. Wilson, Multi-objective influence diagrams, in: Proceesings of the 28th Conference on Uncertainty in Artificial Intelligence (UAI), 2012, pp. 574–583.

[8] R. Marinescu, A. Razak, N. Wilson, Multi-objective constraint optimization with tradeoffs, in: International Conference on Principles and Practice of Constraint Programming (CP), 2013, pp. 497–512.

[9] N. Wilson, An order of magnitude calculus, in: Uncertainty in Artificial Intelligence (UAI), 1995, pp. 548–555.

[10] M. Goldszmidt, J. Pearl, Qualitative probabilities for default reasoning, belief revision, and causal modeling, Artificial Intelligence 84 (1-2) (1996) 57–112.

[11] F. Jensen, V. Jensen, S. Dittmer, From influence diagrams to junction trees, in: Uncertainty in Artificial Intelligence (UAI), 1994, pp. 367–363.

[12] R. Dechter, A new perspective on algorithms for optimizing policies under uncertainty, in: Artificial Intelligence Planning Systems (AIPS), 2000, pp. 72–81.

[13] S. Lauritzen, D. Nilsson, Representing and solving decision problems with limited information, Management Science 47 (9) (2001) 1235–1251.

[14] R. Dechter, Constraint Processing, Morgan Kaufmann Publishers, 2003.

[15] E. Fagiuoli, M. Zaffalon, A note about redundancy in influence diagrams, International Journal of Approximate Reasoning 19 (3-4) (1998) 231–246.

[16] H. Fargier, E. Rollon, N. Wilson, Enabling local computation for partially ordered preferences, Constraints 15 (4) (2010) 516–539.

[17] W. Spohn, Ordinal conditional functions: a dynamic theory of epistemic states, Causation in decision, belief change and statistics 2 (1987) 105–134.

[18] A. Darwiche, M. Goldszmidt, On the relation between kappa calculus and probabilistic reasoning, in: Uncertainty in Artificial Intelligence (UAI), 1994, pp. 145–153.

[19] M. A. Peot, R. D. Shachter, Fusion and proagation with multiple observations in belief networks, Artificial Intelligence 48 (1992) 299–318.

[20] P. Shenoy, Valuation-based systems for Bayesian decision analysis, Operations Research 40 (1) (1992) 463–484.

[21] C. Pralet, T. Schiex, G. Verfaillie, Sequential Decision-Making Problems—Representation and Solution, Wiley, 2009.

[22] J. Kohlas, Information Algebras—Generic Structures for Inference, Springer-Verlag, 2003.

[23] G. R. Shafer, P. Shenoy, Probability propagation, Annals of Mathematics and Artificial Intelligence 2 (1990) 327–352.

[24] P. Shenoy, Valuation network representation and solution of asymmetric decision problems, European Journal of Operational Research 121 (3) (2000) 579–608.

[25] P. Perny, O. Spanjaard, P. Weng, Algebraic Markov Decision Processes, in: International Joint Conference on Artificial Intelligence (IJCAI), 2005, pp. 1372–1377.

[26] D. Kikuti, F. Cozman, Influence diagrams with partially ordered preferences, in: Proceedings of 3rd Multidisciplinary Workshop on Advances in Preference Handling, 2007, pp. 1–7.

[27] D. Kikuti, F. Cozman, R. Filho, Sequential decision making with partially ordered preferences, Artificial Intelligence 175 (7-8) (2011) 1346–1365.

[28] C. de Campos, Q. Ji, Strategy selection in influence diagrams using imprecise probabilities, in: Uncertainty in Artificial Intelligence (UAI), 2008, pp. 121–128.

[29] F. Cozman, Credal networks, Artificiall Intelligence 120 (2) (2000) 199–233.

[30] R. Haenni, Ordered valuation algebras: A generic framework for approximate inference, International Journal of Approximate Reasoning (IJAR) 37 (1) (2004) 1–41.

[31] D. Maua, C. de Campos, M. Zaffalon, Solving limited memory influence diagrams, Journal of Artificial Intelligence Research (JAIR) 44 (1) (2012) 97–140.

[32] D. Maua, C. de Campos, M. Zaffalon, Updating credal networks is approximable in polynomial time, International Journal of Approximate Reasoning (IJAR) 53 (8) (2012) 1183–1199.

[33] M. Diehl, Y. Haimes, Influence diagrams with multiple objectives and tradeoff analysis, IEEE Transactions On Systems, Man, and Cybernetics Part A 34 (3) (2004) 293–304.

[34] M. López-Díaz, L. Rodríguez-Muñiz, Influence diagrams with super value nodes involving imprecise information, European Journal of Operational Research 179 (1) (2007) 203–219.

[35] B. Bonet, J. Pearl, Qualitative MDPs and POMDPs: an order-of-magnitude approximation, in: Uncertainty in Artificial Intelligence (UAI), 2002, pp. 61–68.

[36] L. Garcia, R. Sabbadin, Complexity results and algorithms for possibilistic influence diagrams, Artificial Intelligence 172 (1) (2008) 1018–1044.

[37] R. Cabanas, A. Antonucci, A. Cano, M. Gomez-Olmedo, Evaluating interval-valued influence diagrams, International Journal of Approximate Reasoning 80 (1) (2017) 393–411.

[38] R. Marinescu, R. Dechter, AND/OR branch-and-bound search for combinatorial optimization in graphical models, Artificial Intelligence 173 (16-17) (2009) 1457–1491.

[39] R. Marinescu, R. Dechter, Memory intensive AND/OR search for combinatorial optimization in graphical models, Artificial Intelligence 173 (16-17) (2009) 1492–1524.

Appendix

Part A: Proofs of Results in Section 3

Recall that any $a \in \mathcal{O}$ can be written as $\langle \sigma(a), \hat{a} \rangle$. The following lemma is useful in proving associativity and distributivity results later.

Lemma 22. *Operations $\otimes$ and $\oplus$ on the set of signs $\{+, -, \pm\}$ (as defined in Definitions 7 and 8) are commutative and associative, and $\otimes$ distributes over $\oplus$, i.e., for all $\sigma, \tau, \theta \in \{+, -, \pm\}$, $(\sigma \oplus \tau) \otimes \theta = (\sigma \otimes \theta) \oplus (\tau \otimes \theta)$.*

Proof: Recall that $\otimes$ is commutative and $+ \otimes - = -$, $+ \otimes + = - \otimes - = +$, and $\forall \sigma \in \{+, -, \pm\}$, $\sigma \otimes \pm = \pm$. Also, $+ \oplus + = +$, $- \oplus - = -$, and otherwise, $\sigma \oplus \tau = \pm$.

Associativity of $\otimes$: if any of σ, τ and θ are $\pm$ then $(\sigma \otimes \tau) \otimes \theta = \sigma \otimes (\tau \otimes \theta) = \pm$. Otherwise, σ, τ and θ are all $+$ or $-$. Then, if an even number of σ, τ and θ are $-$, we have $(\sigma \otimes \tau) \otimes \theta = \sigma \otimes (\tau \otimes \theta) = +$. Otherwise, $(\sigma \otimes \tau) \otimes \theta = \sigma \otimes (\tau \otimes \theta) = -$.

Operation $\oplus$ is clearly commutative. Associativity of $\oplus$: If $\sigma = \tau = \theta = +$ then $(\sigma \oplus \tau) \oplus \theta = \sigma \oplus (\tau \oplus \theta) = +$; if $\sigma = \tau = \theta = -$ then $(\sigma \oplus \tau) \oplus \theta = \sigma \oplus (\tau \oplus \theta) = -$; otherwise, $(\sigma \oplus \tau) \oplus \theta = \sigma \oplus (\tau \oplus \theta) = \pm$.

Distributivity: if any of σ, τ and θ are $\pm$ then $(\sigma \oplus \tau) \otimes \theta = (\sigma \otimes \theta) \oplus (\tau \otimes \theta) = \pm$. Otherwise, σ, τ and θ are all in $\{+, -\}$. If $\sigma = \tau$ then $(\sigma \oplus \tau) \otimes \theta = \sigma \otimes \theta = (\sigma \otimes \theta) \oplus (\tau \otimes \theta)$. Else, $\sigma \neq \tau$, and $(\sigma \oplus \tau) \otimes \theta = (\sigma \otimes \theta) \oplus (\tau \otimes \theta) = \pm$. In all cases, we thus have $(\sigma \oplus \tau) \otimes \theta = (\sigma \otimes \theta) \oplus (\tau \otimes \theta)$. $\square$

Below we give some useful properties of addition on $\mathcal{O}$, which follow immediately from the definition.

Lemma 23. *Addition on $\mathcal{O}$ is idempotent: for all $a \in \mathcal{O}$, $a + a = a$. If $\hat{a} < \hat{b}$ then $a + b = a$; if $\hat{a} > \hat{b}$ then $a + b = b$. If $\hat{a} = \hat{b}$ and $a \neq b$ then $a + b = \langle \pm, \hat{a} \rangle$.*

The result below sums up some of the basic algebraic properties of the operations.

Lemma 24. *(i) Multiplication $\times$ on $\mathcal{O}$ is associative and commutative, and for all $a \in \mathcal{O}$, $a \times 0 = 0$ and $a \times 1 = a$.*

(ii) Addition $+$ on $\mathcal{O}$ is associative and commutative, and $a + 0 = a$, for all $a \in \mathcal{O}$.

(iii) For all $a, b, c \in \mathcal{O}$, $(a + b) \times c = a \times c + b \times c$.

Proof: (i) Commutativity of $\times$ follows immediately. Regarding associativity, consider any $a, b, c \in \mathcal{O}$. $(a \times b) \times c = \langle(\sigma(a) \otimes \sigma(b)) \otimes \sigma(c), \hat{a} + \hat{b} + \hat{c}\rangle$. By associativity of multiplication $\otimes$ on signs (Lemma 22), $(\sigma(a) \otimes \sigma(b)) \otimes \sigma(c) = \sigma(a) \otimes (\sigma(b) \otimes \sigma(c))$. Thus, $(a \times b) \times c \otimes \langle\sigma(a) \otimes (\sigma(b) \times \sigma(c)), \hat{a} + \hat{b} + \hat{c}\rangle = a \times (b \times c)$, proving associativity of $\times$.

We have that $a \times 0$, i.e., $a \times \langle\pm, \infty\rangle$ equals $\langle\pm, \infty\rangle = 0$. Also, $a \times 1$, i.e., $\langle\sigma(a), \hat{a}\rangle \times \langle+, 0\rangle$, equals $\langle\sigma(a), \hat{a}\rangle = a$.

(ii): Operation $\oplus$ on signs is commutative, which implies that addition on $\mathcal{O}$ is commutative.

Associativity: Consider any elements a_1, a_2 and a_3 of $\mathcal{O}$, and write $a_i = \langle\sigma_i, m_i\rangle$, for $i = 1, 2, 3$. Let $m = \min\{m_1, m_2, m_3\}$. Case (I): there exists a unique i such that $m_i = m$. Then $(a_1 + a_2) + a_3 = a_1 + (a_2 + a_3) = \langle\sigma_i, m_i\rangle$. Case (II): there exist two values j and k of i such that $m_i = m$. Then $(a_1 + a_2) + a_3 = a_1 + (a_2 + a_3) = \langle\sigma_j, m_j\rangle + \langle\sigma_k, m_k\rangle$. Otherwise, we have Case (III), when $m_1 = m_2 = m_3 = m$. Then, $(a_1 + a_2) + a_3 = \langle(\sigma_1 \oplus \sigma_2) \oplus \sigma_3, m\rangle$, which, since operation $\oplus$ is associative (Lemma 22), is equal to $\langle\sigma_1 \oplus (\sigma_2 \oplus \sigma_3), m\rangle = a_1 + (a_2 + a_3)$. Therefore, in all cases, $(a_1 + a_2) + a_3 = a_1 + (a_2 + a_3)$, proving associativity.

Consider any $a = \langle\sigma, m\rangle \in \mathcal{O}$. If $m = \infty$ then $\sigma = \pm$, and $a = 0$, so $a + 0 = \langle\pm, \infty\rangle + \langle\pm, \infty\rangle = \langle\pm, \infty\rangle = 0 = a$. Otherwise, $m < \infty$ and $\langle\sigma, m\rangle + \langle 0, \infty\rangle = \langle\sigma, m\rangle = a$. Thus, in either case, $a + 0 = a$.

(iii) We will show that for all $a, b, c \in \mathcal{O}$, $(a + b) \times c = a \times c + b \times c$. Case (I): when $\hat{a} < \hat{b}$. Then $\widehat{a \times c} = \hat{a} + \hat{c} < \hat{b} + \hat{c} = \widehat{b \times c}$, so $(a + b) \times c = a \times c = a \times c + b \times c$, using Lemma 23. Case (II): when $\hat{a} > \hat{b}$. Similarly, we obtain $(a + b) \times c = b \times c = a \times c + b \times c$. Case (III): when $\hat{a} = \hat{b}$, equals m, say. Write $(a + b) \times c$ as d, and write $a \times c + b \times c$ as e. Then $\hat{d} = m + \hat{c} = \widehat{a \times c} = \widehat{b \times c} = \hat{e}$. Also, $\sigma(d) = \sigma((a + b) \times c) = (\sigma(a) \oplus \sigma(b)) \otimes \sigma(c)$, which, by Lemma 22, is equal to $(\sigma(a) \otimes \sigma(c)) \oplus (\sigma(b) \otimes \sigma(c))$, which equals $\sigma(e)$, so $\sigma(d) = \sigma(e)$ and $\hat{d} = \hat{e}$, and thus, $d = e$. Hence, $(a + b) \times c = a \times c + b \times c$ in this case also.

Lemma 25. *(i) For all $p, q \in \mathcal{O}_+ \cup \{0\}$, $p + q = 0$ if and only if $p = q = 0$;*

(ii) for all $q \in \mathcal{O}_+$, q^{-1} is the unique element of $\mathcal{O}_+$ with $q \times q^{-1} = 1$.

Proof: (i): Using the fact that 0 is the additive identity element (Lemma 24), we have the following. If $p = q = 0$ then clearly, $p + q = 0$. If $p \neq 0$ and $q = 0$ then $p + q = p \neq 0$. If $p = 0$ and $q \neq 0$ then $p + q = q \neq 0$. To complete the equivalence we just need to show that if p and q are both not equal to 0 then $p + q \neq 0$. But if $p, q \neq 0$ then $\hat{p}, \hat{q} \neq \infty$, so $\widehat{p + q} \neq \infty$, and thus, $p + q \neq 0$.

(ii): Since $q \in \mathcal{O}_+$, $q = \langle +, m \rangle$ for some $m \in \mathbb{Z}$. Consider any $p = \langle +, n \rangle \in \mathcal{O}_+$. Then $q \times p = \langle +, m+n \rangle$, so $q \times p = 1 = \langle +, 0 \rangle$ if and only if $n = -m$, i.e., if and only if $p = q^{-1}$. $\square$

The next result follows immediately from Lemmas 24 and 25.

Lemma 26. *Tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1 \rangle$ is an uncertainty values structure and $\langle \mathcal{O}, +, 0 \rangle$ is a utility values structure.*

The following lemma, which is an immediate consequence of Definition 9, gives a different way of understanding the relation $\succ$.

Lemma 27. *(i) Assume $\hat{a} < \hat{b}$. Then $a \succ b \iff \sigma(a) = +$.*

(ii) Assume $\hat{a} > \hat{b}$. Then $a \succ b \iff \sigma(b) = -$.

(iii) Assume $\hat{a} = \hat{b}$. Then $a \succ b$ if and only if either [$\sigma(a) = +$ and $\sigma(b) \neq +$] or $\sigma(a) = \pm$ and $\sigma(b) = -$].

Lemma 28. *Relation $\succeq$ on $\mathcal{O}$ is a partial order, which is a total order when restricted to $(\mathcal{O} - \mathcal{O}_\pm) \cup \{0\}$.*

Proof: We first show that $\succeq$ is a partial order, i.e., it is reflexive, anti-symmetric and transitive. By definition, $\succeq$ is reflexive. By considering each case in turn, it can be seen that if $a \succ b$ then it is not the case that $b \succ a$. Thus if $a \succeq b$ and $b \succeq a$ then $a = b$, proving anti-symmetry of $\succeq$.

We next prove that $\succeq$ is transitive. Suppose that $a \succeq b$ and $b \succeq c$, for $a, b, c \in \mathcal{O}$. It trivially follows that $a \succeq c$ if $a = b$ or $b = c$, so we can assume that $a \succ b$ and $b \succ c$. There are a number of cases.

Case (I): First consider if $\sigma(a) = +$. If $\sigma(c) = -$ then $a \succ c$. If $\sigma(c) = \pm$ then $b \succ c$ implies that $\sigma(b) = +$ and $\hat{b} \leq \hat{c}$. Then $a \succ b$ implies that $\sigma(a) = +$ and $\hat{a} < \hat{b}$, so $\hat{a} < \hat{c}$, implying that $a \succ c$. If $\sigma(c) = +$ then we have $\sigma(b) = +$ and $\hat{a} < \hat{b} < \hat{c}$, and so, $a \succ c$.

Case (II): Now consider if $\sigma(a) = \pm$. Then $\sigma(b) = -$ and $\hat{a} \geq \hat{b}$. Also, $b \succ c$ implies that $\sigma(c) = -$ and $\hat{b} > \hat{c}$, so $\hat{a} > \hat{c}$ implying $a \succ c$.

Case (III): The final case is when $\sigma(a) = -$. Then $a \succ b$ implies that $\sigma(b) = -$ and $\hat{a} > \hat{b}$. Similarly, $b \succ c$ implies that $\sigma(c) = -$ and $\hat{b} > \hat{c}$. Therefore $\hat{a} > \hat{c}$ and so $a \succ c$ also in this case.

To prove the last part, we need to show that $\succeq$ restricted to $(\mathcal{O} - \mathcal{O}_\pm) \cup \{0\}$ is complete, i.e., for any $a, b \in (\mathcal{O} - \mathcal{O}_\pm) \cup \{0\}$, either $a \succeq b$ or $b \succeq a$. If $\sigma(a) = +$

then $a \succeq b$ unless $\sigma(b) = +$ and $\hat{a} > \hat{b}$, in which case $b \succeq a$. If $\sigma(a) = -$ then $b \succeq a$ unless $\sigma(b) = -$ and $\hat{a} > \hat{b}$, in which case, $a \succeq b$. Thus, in any case, either $a \succeq b$ or $b \succeq a$. $\square$

The following lemma gives two monotonicity properties.

Lemma 29. *Let a, b, c be arbitrary elements of $\mathcal{O}$ such that $a \succeq b$. It follows that $a + c \succeq b + c$, and, if $c \in \mathcal{O}_+ \cup \{0\}$ then $a \times c \succeq b \times c$.*

Proof: Assume $a \succeq b$. We first prove that $a \times c \succeq b \times c$ holds if $c \in \mathcal{O}_+ \cup \{0\}$. If $a = b$ then $a \times c \succeq b \times c$, so we can assume that $a \succ b$. If $c = 0$ then $a \times c = b \times c = 0$, so $a \times c \succeq b \times c$ holds. We can thus assume that $c \neq 0$. Then $\sigma(c) = +$ so $\sigma(a \times c) = \sigma(a)$ and $\sigma(b \times c) = \sigma(b)$. Also, $\widehat{a \times c} = \hat{a} + \hat{c}$ and $\widehat{b \times c} = \hat{b} + \hat{c}$. Lemma 27 then implies that $a \succ b$ if and only if $a \times c \succ b \times c$, so we have $a \times c \succeq b \times c$, as required.

We now prove that $a + c \succeq b + c$ for any $c \in \mathcal{O}$. If $a = b$ then it trivially follows that $a + c \succeq b + c$, by reflexivity of $\succeq$, so we can assume that $a \succ b$. If $\hat{c} < \hat{a}, \hat{b}$ then $a + c = b + c = c$, so $a + c \succeq b + c$ holds. If, on the other hand, $\hat{c} > \hat{a}, \hat{b}$ then, using Lemma 23, $a + c = a$ and $b + c = b$ so again, $a + c \succeq b + c$ holds. We can therefore assume that it is not the case that $\hat{c} < \hat{a}, \hat{b}$, and it is not the case that $\hat{c} > \hat{a}, \hat{b}$.

We consider three cases. (I): Assume $\hat{a} < \hat{b}$. Lemma 27(i) then implies that $\sigma(a) = +$. We have $\hat{a} \leq \hat{c}$ (else, we'd have $\hat{c} > \hat{a}, \hat{b}$). If $\hat{a} < \hat{c}$ then $a + c = a$, and $\widehat{a + c} = \hat{a} < \widehat{b + c} = \min(\hat{b}, \hat{c})$, so $a + c \succ b + c$, by Lemma 27(i). Otherwise, we have $\hat{a} = \hat{c}$. Then, $\hat{c} < \hat{b}$, so $b + c = c$. Also, $\widehat{a + c} = \hat{a} = \hat{c} = \widehat{b + c}$. If $\sigma(c) = +$ or $\sigma(c) = \pm$ then $a + c = c$ so $a + c \succeq b + c$; else, $\sigma(c) = -$, and $a + c = \langle \pm, \hat{c} \rangle \succ \langle -, \hat{c} \rangle = b + c$, so $a + c \succeq b + c$ again.

Case (II): $\hat{a} > \hat{b}$. Then, by Lemma 27(ii), $\sigma(b) = -$. We have that $\hat{b} \leq \hat{c}$ (else $\hat{c} > \hat{a}, \hat{b}$). If $\hat{b} < \hat{c}$ then $b + c = b$ and $\widehat{b + c} = \hat{b} < \widehat{a + c}$, so $a + c \succ b + c$ using Lemma 27(ii). Otherwise, $\hat{b} = \hat{c}$. Then $\hat{c} < \hat{a}$ so $a + c = c$, and $\widehat{a + c} = \hat{c} = \widehat{b + c}$. If $\sigma(c) = \pm$ or $\sigma(c) = -$ then $b + c = c$, so $a + c = b + c$, so $a + c \succeq b + c$; else we have $\sigma(c) = +$, and $b + c = \langle \pm, \hat{c} \rangle \prec \langle +, \hat{c} \rangle = a + c$, so $a + c \succeq b + c$ again.

Case (III): $\hat{a} = \hat{b}$, equals m, say. Then $\hat{c} = m$ (else $\hat{c} < \hat{a}, \hat{b}$ or $\hat{c} > \hat{a}, \hat{b}$), and therefore, $\widehat{a + c} = \widehat{b + c} = m$. Suppose, to prove a contradiction, that $a + c \not\succeq b + c$. Using Lemma 27(iii), either (A) $\sigma(a + c) = \pm$ and $\sigma(b + c) = +$ or (B) $\sigma(a + c) = -$ and $\sigma(b + c) \neq -$. If (A) holds then $\sigma(b) = \sigma(c) = +$, and $\sigma(a) \neq +$, which contradicts $a \succ b$. If (B) holds then $\sigma(a + c) = -$ so

$\sigma(a) = \sigma(c) = -$ and $\sigma(b) \neq -$, which again contradicts $a \succ b$. Thus in every case, we have $a + c \succeq b + c$, as required. $\square$

We define an u.u.v. structure $\mathfrak{U}$ for the order of magnitude case to be the tuple $\langle Q, +_Q, \times_Q, 0_Q, 1_Q, U, +_U, 0_U, \times_{QU} \rangle$, where U is $\mathcal{O}$ and Q is $\mathcal{O}_+ \cup \{0\}$. It can be shown that the previously stated properties including ($*1$), ($*2$), ($*3$), and ($*4$) all hold, and the operations respect the ordering $\succeq$.

Proposition 1. *Define $\mathfrak{U}^{\mathcal{O}}$ to be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}, +, 0, \times \rangle$. Then $\mathfrak{U}^{\mathcal{O}}$ is an uncertainty-utility values structure that respects partial order $\succeq$, i.e., $\succeq$ is respected by $+$ and the operation $\times : \mathcal{O}_+ \cup \{0\} \times \mathcal{O} \to \mathcal{O}$.*

Proof: Lemma 26 shows that $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1 \rangle$ is an uncertainty values structure and $\langle \mathcal{O}, +, 0 \rangle$ is a utility values structure. Lemma 24(i) and (iii) imply that $\times$, when considered as a function from $\mathcal{O}_+ \cup \{0\} \times \mathcal{O}$ to $\mathcal{O}$, satisfies properties ($*1$), ($*2$), ($*3$), and ($*4$), and so $\mathfrak{U}^{\mathcal{O}}$ is an uncertainty-utility values structure. Lemma 28 shows that $\succeq$ is a partial order, and Lemma 29 shows that the two operations respect $\succeq$. $\square$

Lemma 30. *Operations $\boxplus_*$ and $\boxplus^*$ on the set of signs $\{+, -\}$ (as defined in Definitions 10 and 11) are commutative and associative.*

Proof: Commutativity of the two operations follows immediately from the definitions. For $\sigma, \tau, \theta \in \{+, -\}$, $(\sigma \boxplus_* \tau) \boxplus_* \theta$ is equal to $+$ if and only if $\sigma = \tau = \theta = +$, and the same holds for $\sigma \boxplus_* (\tau \boxplus_* \theta)$. Thus $(\sigma \boxplus_* \tau) \boxplus_* \theta = \sigma \boxplus_* (\tau \boxplus_* \theta)$. Similarly, $(\sigma \boxplus^* \tau) \boxplus^* \theta$ and $\sigma \boxplus^* (\tau \boxplus^* \theta)$ are both equal to $-$ if and only if $\sigma = \tau = \theta = -$, and so $(\sigma \boxplus^* \tau) \boxplus^* \theta = \sigma \boxplus^* (\tau \boxplus^* \theta)$. $\square$

Lemma 31. *(i) Operations $+_*$ and $+^*$ on $\mathcal{O}'$ are idempotent, associative and commutative, and $a +_* 0 = a +^* 0 = a$, for all $a \in \mathcal{O}'$.*

(ii) For $a \in \mathcal{O}'$ and $p, q \in \mathcal{O}_+ \cup \{0\}$, $(p + q) \times a = (p \times a) +_* (q \times a) = (p \times a) +^* (q \times a)$;

(iii) For $a, b \in \mathcal{O}'$ and $p \in \mathcal{O}_+ \cup \{0\}$. $p \times (a +_* b) = (p \times a) +_* (p \times b)$; $p \times (a +^* b) = (p \times a) +^* (p \times b)$;

Proof: (i): The definitions immediately imply that the idempotency of $+_*$ and $+^*$, and that $a +_* 0 = a +^* 0 = a$, for all $a \in \mathcal{O}'$. Operations $\boxplus_*$ and $\boxplus^*$ on signs are commutative, which implies that $+_*$ and $+^*$ are commutative.

Associativity: Consider any elements a_1, a_2 and a_3 of $\mathcal{O}'$, and write $a_i = \langle \sigma_i, m_i \rangle$, for $i = 1, 2, 3$. Let $m = \min\{m_1, m_2, m_3\}$. Case (I): there exists a unique i such that $m_i = m$. Then $(a_1 +_* a_2) +_* a_3 = a_1 +_* (a_2 +_* a_3) = \langle \sigma_i, m_i \rangle$, and $(a_1 +^* a_2) +^* a_3 = a_1 +^* (a_2 +^* a_3) = \langle \sigma_i, m_i \rangle$. Case (II): there exist two values j and k of i such that $m_i = m$. Then $(a_1 +_* a_2) +_* a_3 = a_1 +_* (a_2 +_* a_3) = \langle \sigma_j, m_j \rangle +_* \langle \sigma_k, m_k \rangle$, and $(a_1 +^* a_2) +^* a_3 = a_1 +^* (a_2 +^* a_3) = \langle \sigma_j, m_j \rangle +^* \langle \sigma_k, m_k \rangle$. Otherwise, we have Case (III), when $m_1 = m_2 = m_3 = m$. Then, $(a_1 +_* a_2) +_* a_3 = \langle (\sigma_1 \boxplus_* \sigma_2) \boxplus_* \sigma_3, m \rangle$, which, since operation $\boxplus_*$ is associative (Lemma 30), is equal to $\langle \sigma_1 \boxplus_* (\sigma_2 \boxplus_* \sigma_3), m \rangle = a_1 +_* (a_2 +_* a_3)$. Similarly, the associativity of $\boxplus^*$ implies that $(a_1 +^* a_2) +^* a_3 = a_1 +^* (a_2 +^* a_3)$. Therefore, in all cases, $(a_1 +_* a_2) +_* a_3 = a_1 +_* (a_2 +_* a_3)$, and $(a_1 +^* a_2) +^* a_3 = a_1 +^* (a_2 +^* a_3)$ proving the two associativity properties.

(ii): Consider any $a \in \mathcal{O}'$ and $p, q \in \mathcal{O}_+ \cup \{0\}$. If $p = q$ then idempotency of $+$, $+_*$ and $+^*$ implies $(p + q) \times a = (p \times a) +_* (q \times a) = (p \times a) +^* (q \times a)$. Now suppose $p \neq q$, so $\hat{p} \neq \hat{q}$. Without loss of generality (because of commutativity), we can assume $\hat{p} < \hat{q}$. Then $p + q = p$. Also, $\widehat{p \times a} = \hat{p} + \hat{a} < \hat{q} + \hat{a} = \widehat{q \times a}$, so $(p \times a) +_* (q \times a) = (p \times a) +^* (q \times a) = p \times a = (p + q) \times a$.

(iii): Consider any $a, b \in \mathcal{O}'$ and $p \in \mathcal{O}_+ \cup \{0\}$. The equalities clearly hold if $p = 0$, by nilpotence of 0, so we can assume that $p \neq 0$. Commutativity means that without loss of generality we can assume that $\hat{a} \leq \hat{b}$. We first consider the case when $\hat{a} < \hat{b}$. Then also $\widehat{p \times a} < \widehat{p \times b}$, so $(p \times a) +_* (p \times b) = p \times a = p \times (a +_* b)$, and $(p \times a) +^* (p \times b) = p \times a = p \times (a +^* b)$. Now consider the case when $\hat{a} = \hat{b}$ (equals m, say), so also $\widehat{p \times a} = \widehat{p \times b} = \hat{p} + m$. Let $d = p \times (a +_* b)$, and let $e = (p \times a) +_* (p \times b)$. Then $\hat{d} = \hat{p} + m = \hat{e}$. Also, $\sigma(p \times a) = \sigma(a)$, and $\sigma(p \times b) = \sigma(b)$, so $\sigma(e) = \sigma(a) \boxplus_* \sigma(b) = \sigma(d)$. Since $\hat{d} = \hat{e}$ and $\sigma(d) = \sigma(e)$, $d = e$, i.e., $p \times (a +_* b) = (p \times a) +_* (p \times b)$. The same argument can be used to prove that $p \times (a +^* b) = (p \times a) +^* (p \times b)$. $\square$

Lemma 32. *Let a, b, c be arbitrary elements of $\mathcal{O}'$ such that $a \succeq b$. Then, $a +_* c \succeq b +_* c$, and $a +^* c \succeq b +^* c$.*

Proof: Assume that $a \succeq b$. We will prove that $a +_* c \succeq b +_* c$, and $a +^* c \succeq b +^* c$ for any $c \in \mathcal{O}'$. If $a = b$ then it trivially follows that $a +_* c \succeq b +_* c$, and $a +^* c \succeq b +^* c$ so we can assume that $a \succ b$. If $\hat{c} < \hat{a}, \hat{b}$ then $a +_* c = b +_* c = c$, so $a +_* c \succeq b +_* c$ holds, and similarly, $a +^* c = b +^* c = c$, and $a +^* c \succeq b +^* c$ holds. Another easy case is if $\hat{c} > \hat{a}, \hat{b}$, for then $a +_* c = a +^* c = a$ and $b +_* c = b +^* c = b$ so again, $a +_* c \succeq b +_* c$ and $a +^* c \succeq b +^* c$ hold. We can therefore assume that it is not the case that $\hat{c} < \hat{a}, \hat{b}$, and it is not the case that $\hat{c} > \hat{a}, \hat{b}$.

We consider three cases, similarly to the proof of Lemma 29. (I): Assume $\hat{a} < \hat{b}$. Lemma 27(i) then implies that $\sigma(a) = +$. We have $\hat{a} \leq \hat{c}$ (else, we'd have $\hat{c} > \hat{a}, \hat{b}$). If $\hat{a} < \hat{c}$ then $a +_* c = a$, and $\widehat{a +_* c} = \hat{a} < \widehat{b +_* c} = \min(\hat{b}, \hat{c})$, so $a +_* c \succ b +_* c$, by Lemma 27(i), with the same argument also showing that $a +^* c \succ b +^* c$. Otherwise, we have $\hat{a} = \hat{c}$. Then, $\hat{c} < \hat{b}$, so $b +_* c = b +^* c = c$. Also, $\widehat{a +_* c} = \hat{a} = \hat{c} = \widehat{b +_* c}$, and similarly, $\widehat{a +^* c} = \widehat{b +^* c}$. If $\sigma(c) = +$ then $a +_* c = a +^* c = c$ so $a +_* c \succeq b +_* c$ and $a +^* c \succeq b +^* c$; else, $\sigma(c) = -$; then we have $a +_* c = b +_* c = c$ so $a +_* c \succeq b +_* c$ again; also, $a +^* c = \langle +, \hat{c} \rangle \succ \langle -, \hat{c} \rangle = b +^* c$, so $a +^* c \succeq b +^* c$.

Case (II): $\hat{a} > \hat{b}$. Then, by Lemma 27(ii), $\sigma(b) = -$. We have that $\hat{b} \leq \hat{c}$ (else $\hat{c} > \hat{a}, \hat{b}$). If $\hat{b} < \hat{c}$ then $b +_* c = b +^* c = b$ and $\widehat{b +_* c} = \hat{b} < \widehat{a +_* c}$, and $\widehat{b +^* c} = \hat{b} < \widehat{a +^* c}$, so $a +_* c \succ b +_* c$ and $a +^* c \succ b +^* c$, using Lemma 27(ii). Otherwise, $\hat{b} = \hat{c}$. Then $\hat{c} < \hat{a}$ so $a +_* c = a +^* c = c$, and $\widehat{a +_* c} = \widehat{a +^* c} = \hat{c} = \widehat{b +_* c} = \widehat{b +^* c}$. If $\sigma(c) = -$ then $b +_* c = b +^* c = c$, so $a +_* c = b +_* c$ and $a +^* c = b +^* c$, so $a +_* c \succeq b +_* c$ and $a +^* c \succeq b +^* c$. Else we have $\sigma(c) = +$. Then, $b +_* c = \langle -, \hat{c} \rangle \prec \langle +, \hat{c} \rangle = a +_* c$, so $a +_* c \succeq b +_* c$. Also, $b +^* c = c = a +^* c$ so $a +^* c \succeq b +^* c$ again.

Case (III): $\hat{a} = \hat{b}$, equals m, say. Then $\hat{c} = m$ (else $\hat{c} < \hat{a}, \hat{b}$ or $\hat{c} > \hat{a}, \hat{b}$), and so $\widehat{a +_* c} = \widehat{b +_* c} = m$, and $\widehat{a +^* c} = \widehat{b +^* c} = m$. Suppose, to prove a contradiction, that $a +_* c \not\succeq b +_* c$. Then $\sigma(a +_* c) = -$ and $\sigma(b +_* c) = +$. This implies $\sigma(b) = \sigma(c) = +$, which contradicts $a \succ b$. The case for $+^*$ is similar: suppose, to prove a contradiction, that $a +^* c \not\succeq b +^* c$. Then $\sigma(a +^* c) = -$ and $\sigma(b +^* c) = +$. Then $\sigma(a) = \sigma(c) = -$ which contradicts $a \succ b$. Thus, in all cases we have $a +_* c \succeq b +_* c$, and $a +^* c \succeq b +^* c$. $\square$

Proposition 15. *Define $\mathfrak{U}_L^{\mathcal{O}'}$ to be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}', +_*, 0, \times \rangle$, and define $\mathfrak{U}_U^{\mathcal{O}'}$ to be the tuple $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1, \mathcal{O}', +^*, 0, \times \rangle$. Then $\mathfrak{U}_L^{\mathcal{O}'}$ and $\mathfrak{U}_U^{\mathcal{O}'}$ are both uncertainty-utility values structures that respect total order $\succeq$.*

Proof: Lemma 26 shows that $\langle \mathcal{O}_+ \cup \{0\}, +, \times, 0, 1 \rangle$ is an uncertainty values structure, and Lemma 31(i) implies that $\langle \mathcal{O}', +_*, 0 \rangle$ and $\langle \mathcal{O}', +^*, 0 \rangle$ are utility values structures. Lemma 24(i) and Lemma 31(ii) and (iii) imply that $\times$, when considered as a function from $\mathcal{O}_+ \cup \{0\} \times \mathcal{O}'$ to $\mathcal{O}'$, satisfies properties ($*1$), ($*2$), ($*3$), and ($*4$) with respect to both $+_*$ and $+^*$, and so $\mathfrak{U}_L^{\mathcal{O}'}$ and $\mathfrak{U}_U^{\mathcal{O}'}$ are both uncertainty-utility values structures. Lemma 28 shows that $\succeq$ restricted to $\mathcal{O}'$ is a total order, and Lemma 32 and Lemma 29 imply that the two operations in both uncertainty-utility values structures respect $\succeq$. $\square$

Part B: Proving Proposition 2

In this section we develop results which enable the proof of Proposition 2.

Proposition 2. *Let $\mathfrak{I} = \langle G, (\Phi, \Psi)\rangle$ be an ID-system over $\mathfrak{U}$, and let τ be a legal elimination sequence for $\mathfrak{I}$. Let τ_k be the part of τ starting just after D_k, i.e., if τ is $Y_1, \ldots, Y_n$ and $Y_j = D_k$ then τ_k is the sequence $Y_{k+1}, \ldots, Y_n$. Let (Φ_k, Ψ_k) be $\mathbb{M}^{+,\vee}_{\tau_k}\big(\Phi, \Psi\big)$ (where if τ_k is empty, $\Phi_k = \Phi$ and $\Psi_k = \Psi$). Then, $\mathbb{M}^{+,\vee}_{\tau}\big(\Phi, \Psi\big)$ is well defined, that is, for each $k = m, \ldots, 1$ and each $\mathbf{P} \in \Phi_k$, $\mathbf{P}$ does not depend on D_k.*

The following result gives the distributivity properties for uncertainty values. This is used to prove Lemma 34.

Lemma 33. *Let $\langle Q, +, \times, 0, 1\rangle$ be an uncertainty values structure, let $p \in Q$, and for each $i \in I$ and $j \in J$, let p_i and q_j be elements of Q. Then the following hold.*

(i) $p \times (\sum_{j\in J} q_j) = \sum_{j\in J} p \times q_j$.

(ii) $(\sum_{i\in I} p_i) \times (\sum_{j\in J} q_j) = \sum_{i\in I, j\in J} p_i \times q_j$.

(iii) For each $j = 1, \ldots, h$ and $i \in I_j$, let $p_{j,i}$ be an element of Q. Then

$$\prod_{j=1}^{h}(\sum_{i_j\in I_j} p_{j,i_j}) = \sum_{i_1\in I_1,\ldots,i_h\in I_h} \prod_{j=1}^{h} p_{j,i_j}.$$

Proof: (i) follows by iterative use of distributivity. For, (ii), $(\sum_{i\in I} p_i)\times(\sum_{j\in J} q_j) = \sum_{j\in J}((\sum_{i\in I} p_i) \times q_j)$, using (i), which equals $\sum_{j\in J}\sum_{i\in I}(p_i \times q_j)$, using commutativity of $\times$ and (i) again. This equals $\sum_{i\in I, j\in J} p_i \times q_j$.
(iii): We prove this by induction on h. It clearly holds for $h = 1$. Now suppose it holds for $h-1$ (where $h > 1$). We will show that it holds for h.

$$\prod_{j=1}^{h}(\sum_{i_j\in I_j} p_{j,i_j}) = \left(\prod_{j=1}^{h-1}(\sum_{i_j\in I_j} p_{j,i_j})\right) \times (\sum_{i_h\in I_h} p_{h,i_h}).$$

By the inductive hypothesis, this equals

$$\left(\sum_{i_1 \in I_1, \ldots, i_{h-1} \in I_{h-1}} \prod_{j=1}^{h-1} p_{j,i_j}\right) \times (\sum_{i_h \in I_h} p_{h,i_h}) = \sum_{i_1 \in I_1, \ldots, i_{h-1} \in I_{h-1}} \sum_{i_h \in I_h} \prod_{j=1}^{h} p_{j,i_j},$$

using part (ii). This equals $\sum_{i_1 \in I_1, \ldots, i_h \in I_h} \prod_{j=1}^{h} p_{j,i_j}$, completing the inductive proof. □

Lemma 34. *Let $\mathfrak{U}$ be some weak u.u.v. structure extending uncertainty values structure $\langle Q, +, \times, 0, 1\rangle$. For $j = 1, \ldots, h$, let $\mathbf{P}_j$ be an $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$, and let $\mathbf{S}_j \subseteq \mathbf{X}$ be such that for each $j, k \in \{1, \ldots, h\}$, (i) $\mathbf{S}_j \subseteq sc(\mathbf{P}_j)$; and (ii) $\mathbf{S}_j \cap sc(\mathbf{P}_k) = \emptyset$ if $j \neq k$, so, in particular, the sets $\mathbf{S}_j$ are pairwise disjoint. Then*

$$\prod_{j=1}^{h} (\sum_{\mathbf{S}_j} \mathbf{P}_j) = \sum_{\mathbf{S}_1 \cup \cdots \cup \mathbf{S}_h} (\mathbf{P}_1 \times \cdots \times \mathbf{P}_h).$$

Proof: Let $\mathbf{P} = \prod_{j=1}^{h} (\sum_{\mathbf{S}_j} \mathbf{P}_j)$, and let $\mathbf{P}' = \sum_{\mathbf{S}_1 \cup \cdots \cup \mathbf{S}_h} (\mathbf{P}_1 \times \cdots \times \mathbf{P}_h)$. We need to show that $\mathbf{P} = \mathbf{P}'$.

First we show that $\mathbf{P}$ and $\mathbf{P}'$ have the same scope. For $j = 1, \ldots, h$, let $\mathbf{T}_j = sc(\mathbf{P}_j)$, the scope of $\mathbf{P}_j$. Now, $sc(\mathbf{P}) = \bigcup_{j=1}^{h} (\mathbf{T}_j - \mathbf{S}_j)$, and $sc(\mathbf{P}') = \bigcup_{j=1}^{h} \mathbf{T}_j - \bigcup_{j=1}^{h} \mathbf{S}_j$. First suppose that $X \in sc(\mathbf{P})$. Then there exists some $k \in \{1, \ldots, h\}$ such that $X \in \mathbf{T}_k - \mathbf{S}_k$. Therefore $X \in \bigcup_{j=1}^{h} \mathbf{T}_j$. To prove a contradiction, assume that $X \notin sc(\mathbf{P}')$, so there exists some j such that $X \in \mathbf{S}_j$. By the hypothesis, for all $i \neq j$, $\mathbf{T}_i \cap \mathbf{S}_j = \emptyset$, so $X \notin \mathbf{T}_i$ unless $i = j$. This implies that $j = k$, which is a contradiction, since $X \notin \mathbf{S}_k$. Conversely, suppose that $X \in sc(\mathbf{P}')$. This implies that for some $j \in \{1, \ldots, h\}$, $X \in \mathbf{T}_j$. Also, $X \notin \mathbf{S}_j$, so that $X \in \mathbf{T}_j - \mathbf{S}_j$. Hence, $X \in sc(\mathbf{P})$.

Let $\mathbf{R} = sc(\mathbf{P}) = sc(\mathbf{P}')$, and consider any $\mathbf{y} \in \Omega_{\mathbf{R}}$. We need to show that $\mathbf{P}(\mathbf{y}) = \mathbf{P}'(\mathbf{y})$. We have that $\mathbf{P}(\mathbf{y}) = \prod_{j=1}^{h} (\sum_{\mathbf{S}_j} \mathbf{P}_j)(\mathbf{y}) = \prod_{j=1}^{h} (\sum_{\mathbf{s}_j \in \Omega_{\mathbf{S}_j}} \mathbf{P}_j(\mathbf{y}\mathbf{s}_j))$.

Also, $\mathbf{P}'(\mathbf{y}) = \sum_{\mathbf{s}_1 \in \Omega_{\mathbf{S}_1}, \ldots, \mathbf{s}_h \in \Omega_{\mathbf{S}_h}} (\mathbf{P}_1 \times \cdots \times \mathbf{P}_h)(\mathbf{y}\mathbf{s}_1 \cdots \mathbf{s}_h)$. Now, for each $j = 1, \ldots, h$, $\mathbf{P}_j(\mathbf{y}\mathbf{s}_1 \cdots \mathbf{s}_h) = \mathbf{P}_j(\mathbf{y}\mathbf{s}_j)$, since for $k \neq j$, $\mathbf{S}_k \cap sc(\mathbf{P}_j) = \emptyset$. Therefore, $\mathbf{P}'(\mathbf{y}) = \sum_{\mathbf{s}_1 \in \Omega_{\mathbf{S}_1}, \ldots, \mathbf{s}_h \in \Omega_{\mathbf{S}_h}} (\mathbf{P}_1(\mathbf{y}\mathbf{s}_1) \times \cdots \times \mathbf{P}_h(\mathbf{y}\mathbf{s}_h))$. Lemma 33 then implies that $\mathbf{P}(\mathbf{y}) = \mathbf{P}'(\mathbf{y})$. □

This result, which is related to Lemma 3(v), gives situations in which a summation-elimination can be moved inside a combination.

Lemma 35. *Let $\mathbf{P}_1$ and $\mathbf{P}_2$ be $\mathfrak{U}$-uncertainty functions over $\mathbf{X} \cup \mathbf{D}$ and let $\mathbf{S} \subseteq \mathbf{X}$ be such that $\mathbf{S} \subseteq sc(\mathbf{P}_2)$.*

(i) If $\mathbf{S} \cap sc(\mathbf{P}_1) = \emptyset$ then $\sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2) = \mathbf{P}_1 \times \sum_{\mathbf{S}} \mathbf{P}_2$.

(ii) If $\mathbf{P}_1$ is a constant function then there exists a constant function $\mathbf{P}'_1$ with $\sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2) = \mathbf{P}'_1 \times \sum_{\mathbf{S}} \mathbf{P}_2$.

Proof: (i): Let $\mathbf{P} = \sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2)$ and let $\mathbf{P}' = \mathbf{P}_1 \times \sum_{\mathbf{S}} \mathbf{P}_2$. We need to show that $\mathbf{P} = \mathbf{P}'$. First we show that $sc(\mathbf{P}) = sc(\mathbf{P}')$. $sc(\mathbf{P}) = (sc(\mathbf{P}_1) \cup sc(\mathbf{P}_2)) - \mathbf{S} = (sc(\mathbf{P}_1) - \mathbf{S}) \cup (sc(\mathbf{P}_2) - \mathbf{S}) = sc(\mathbf{P}_1) \cup (sc(\mathbf{P}_2) - \mathbf{S})$ (since $\mathbf{S} \cap sc(\mathbf{P}_1) = \emptyset$), which equals $sc(\mathbf{P}')$.

Now consider any assignment $\mathbf{x}$ to $sc(\mathbf{P})$ ($= sc(\mathbf{P}')$). Then, $\mathbf{P}(\mathbf{x}) = \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}}(\mathbf{P}_1 \times \mathbf{P}_2)(\mathbf{xs}) = \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}}(\mathbf{P}_1(\mathbf{x}) \times \mathbf{P}_2(\mathbf{xs})) = \mathbf{P}_1(\mathbf{x}) \times \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}} \mathbf{P}_2(\mathbf{xs})$ (by distributivity), which equals $\mathbf{P}'(\mathbf{x})$, as required.

(ii): Let $\mathbf{T} = sc(\mathbf{P}_1)$. Since $\mathbf{P}_1$ is a constant function, there exists $q \in Q$ such that for all $\mathbf{y} \in \Omega_{\mathbf{T}}$, $\mathbf{P}_1(\mathbf{y}) = q$. Define $\mathbf{P}'_1$ to be the constant function with scope $\mathbf{T} - \mathbf{S}$, taking the value q. $sc(\sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2)) = (\mathbf{T} \cup sc(\mathbf{P}_2)) - \mathbf{S} = (\mathbf{T} - \mathbf{S}) \cup (sc(\mathbf{P}_2) - \mathbf{S})$. This equals $sc(\mathbf{P}'_1 \times \sum_{\mathbf{S}} \mathbf{P}_2)$.

We have $(\sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2))(\mathbf{x}) = \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}}(\mathbf{P}_1(\mathbf{xs}) \times \mathbf{P}_2(\mathbf{xs})) = \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}}(q \times \mathbf{P}_2(\mathbf{xs}))$, which equals $q \times \sum_{\mathbf{s} \in \Omega_{\mathbf{S}}} \mathbf{P}_2(\mathbf{xs}))$ by distributivity. This equals $q \times (\sum_{\mathbf{S}} \mathbf{P}_2)(\mathbf{x})$. Also, $(\mathbf{P}'_1 \times \sum_{\mathbf{S}} \mathbf{P}_2)(\mathbf{x}) = \mathbf{P}'_1(\mathbf{x}) \times (\sum_{\mathbf{S}} \mathbf{P}_2)(\mathbf{x})$, which also equals $q \times (\sum_{\mathbf{S}} \mathbf{P}_2)(\mathbf{x})$. Thus, $\sum_{\mathbf{S}}(\mathbf{P}_1 \times \mathbf{P}_2) = \mathbf{P}'_1 \times \sum_{\mathbf{S}} \mathbf{P}_2$. $\square$

The following lemma gives some basic properties of the sets $\mathbf{I}_j$. In particular, part (ii) shows that if a chance variable is a descendant of a decision variable, then the chance variable appears later in the elimination sequence than the decision variable (so the chance variable is eliminated first).

Lemma 36. *Let $\langle G, (\Phi, \Psi) \rangle$ be an ID-system over $\mathfrak{U}$, with associated decision variables $D_1, \ldots, D_m$, and ordered partition of chance variables $\mathbf{I}_0, \ldots, \mathbf{I}_m$. Let $X \in \mathbf{X}$ be a chance variable. Then the following hold.*

(i) For all $k = 0, \ldots, m$, $X \in \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$ if and only if for all $j = 1, \ldots, k$, $(X, D_j) \notin G$.

(ii) If D_k is a G-ancestor of X then $X \in \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$.

Proof: (i) By the definition, for $j = 0, \ldots, m-1$, $X \in \mathbf{I}_j$ if and only if $(X, D_{j+1}) \in G$ and for all $i \leq j$, $(X, D_i) \notin G$. Also, $X \in \mathbf{I}_m$ if and only if for all i such that

$1 \leq i \leq m$, $(X, D_i) \notin G$. Thus if $X \in \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$ then for all $j = 1, \ldots, k$, $(X, D_j) \notin G$. Conversely, suppose that for all for all $j = 1, \ldots, k$, $(X, D_j) \notin G$. Then for all $j = 1, \ldots, k$, $X \notin \mathbf{I}_{j-1}$, so $X \notin \mathbf{I}_0 \cup \cdots \cup \mathbf{I}_{k-1}$. Thus $X \in \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$, since $X \in \mathbf{X} = \mathbf{I}_0 \cup \cdots \cup \mathbf{I}_m$.

(ii): Suppose, to prove a contradiction, that D_k is a G-ancestor of X, and $X \notin \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$. By part (i), $(X, D_j) \in G$ for some $j \in \{1, \ldots, k\}$. Then $j = k$ or D_j is an ancestor (in fact, parent) of D_k, so, in either case, X is an ancestor of D_k. This contradicts acyclicity of G. $\square$

Lemma 37 below is a key milestone in proving Proposition 2. It is used to show that if each element of Φ^i can be written in a particular way and the next variable to be deleted is a decision variable D, then Φ^i does not depend on D. It is used in proving Property (5) in Proposition 16.

Lemma 37. *Let $\langle G, (\Phi, \Psi) \rangle$ be an ID-system over $\mathfrak{U}$, and let $D \in \mathbf{D}$ be a decision variable. Consider $\mathbf{P}$ of form $\mathbb{C} \times \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$, where $\mathbb{C}$ is a constant function, $\mathbf{S} \subseteq \mathbf{T} \subseteq \mathbf{X}$ and $\mathbf{S}$ contains all G-descendants of D in $\mathbf{T}$. Then $\mathbf{P}$ can be written as $\mathbb{C}' \times \sum_{\mathbf{S}-\mathbf{S}'} \prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X$, where $\mathbb{C}'$ is a constant function, $\mathbf{S}'$ is the set of elements of $\mathbf{T}$ that are G-descendants of D (and thus $\mathbf{S}' \subseteq \mathbf{S}$ and $\mathbf{T} - \mathbf{S}'$ contains no G-descendants of D), and for all $X \in \mathbf{T} - \mathbf{S}'$, $\mathbf{P}_X$ does not involve D, and so $\mathbf{P}$ does not depend on D.*

Proof: Let $\mathbf{S}'$ be the set of elements of $\mathbf{T}$ that are G-descendants of D. Thus, $\mathbf{T}-\mathbf{S}'$ contains no G-descendants of D. By the assumption on $\mathbf{S}$, we have $\mathbf{S}' \subseteq \mathbf{S}$. We can write $\mathbf{P}$ as $\mathbb{C} \times \sum_{\mathbf{S}-\mathbf{S}'} \sum_{\mathbf{S}'} ((\prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X) \times (\prod_{X \in \mathbf{S}'} \mathbf{P}_X))$. Now, suppose that $\mathbf{P}_X$ involves X' for some $X' \in \mathbf{S}'$ and $X \in \mathbf{T}$. Then $(X', X) \in G$ and X' is a descendant of D, so X is a descendant of D, which implies that $X \in \mathbf{S}'$. This shows that for all $X \in \mathbf{T} - \mathbf{S}'$, $sc(\mathbf{P}_X) \cap \mathbf{S}' = \emptyset$. Thus $sc(\prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X) \cap \mathbf{S}' = \emptyset$ and $\mathbf{S}' \subseteq sc(\prod_{X \in \mathbf{S}'} \mathbf{P}_X)$, so we can apply Lemma 35(i) to give that

$$\mathbf{P} = \mathbb{C} \times \sum_{\mathbf{S}-\mathbf{S}'} (\prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X) \sum_{\mathbf{S}'} (\prod_{X \in \mathbf{S}'} \mathbf{P}_X).$$

Let us order $\mathbf{S}'$ as $X_1, \ldots, X_k$ in a way that is compatible with G, so that if $(X_i, X_j) \in G$ then $i < j$. (This is always possible, because G is acyclic.) Thus, if $X_i \in sc(\mathbf{P}_{X_j})$ then either $i = j$ or $(X_i, X_j) \in G$, so in either case, $i \leq j$. This implies that for $i > j$, $X_i \notin sc(\mathbf{P}_{X_j})$. Therefore, $X_k \notin sc(\mathbf{P}_{X_1} \times \cdots \times \mathbf{P}_{X_{k-1}})$. Lemma 35(i) leads to

$$\sum_{X_1} \cdots \sum_{X_k} (\mathbf{P}_{X_1} \times \cdots \times \mathbf{P}_{X_k}) = \sum_{X_1} \cdots \sum_{X_{k-1}} \mathbf{P}_{X_1} \times \cdots \times \mathbf{P}_{X_{k-1}} \sum_{X_k} \mathbf{P}_{X_k}.$$

Iterating this leads to

$$\sum_{\mathbf{S}'}(\prod_{X\in\mathbf{S}'}\mathbf{P}_X) = \sum_{X_1}\cdots\sum_{X_k}(\mathbf{P}_{X_1}\times\cdots\times\mathbf{P}_{X_k}) = \sum_{X_1}\mathbf{P}_{X_1}\sum_{X_2}\mathbf{P}_{X_2}\cdots\sum_{X_k}\mathbf{P}_{X_k}.$$

Now, each $\mathbf{P}_{X_j}$ is a conditional $\mathfrak{U}$-uncertainty function. Thus, in particular, $\sum_{X_k}\mathbf{P}_{X_k}$ equals a constant function, say $\mathbb{C}_k$. Also, $\sum_{X_{k-1}}(\mathbf{P}_{X_{k-1}}\times\mathbb{C}_k)$ equals $\mathbb{C}'_k\times\sum_{X_{k-1}}\mathbf{P}_{X_{k-1}}$ for some constant function $\mathbb{C}'_k$, by Lemma 35(ii). $\sum_{X_{k-1}}\mathbf{P}_{X_{k-1}}$ equals a constant function, say $\mathbb{C}_{k-1}$, and so $\sum_{X_{k-1}}(\mathbf{P}_{X_{k-1}}\times\mathbb{C}_k)$ is a constant function. Iterating this implies that $\sum_{\mathbf{S}'}(\prod_{X\in\mathbf{S}'}\mathbf{P}_X)$ equals a constant function, $\mathbb{C}''$.

Therefore, $\mathbf{P}=\mathbb{C}\times\sum_{\mathbf{S}-\mathbf{S}'}((\prod_{X\in\mathbf{T}-\mathbf{S}'}\mathbf{P}_X)\times\mathbb{C}'')$, which equals $\mathbf{P}=\mathbb{C}'\times\sum_{\mathbf{S}-\mathbf{S}'}(\prod_{X\in\mathbf{T}-\mathbf{S}'}\mathbf{P}_X)$ for some constant function $\mathbb{C}'$, by Lemma 35(ii).

By definition, $\mathbf{P}_X$ involves D if and only if $(D,X)\in G$, i.e., X is a parent of D. Thus if, for some $X\in\mathbf{T}$, $\mathbf{P}_X$ involves D then $X\in\mathbf{S}'$. Therefore, if $X\in\mathbf{T}-\mathbf{S}'$ then $\mathbf{P}_X$ does not involve D. Let $R=sc(\mathbf{P})-\{D\}$, and let $\mathbf{x}$ be any element of Ω_R, and let d and d' be two assignments to D. Then, $\mathbf{P}(\mathbf{x}d)=\mathbf{P}(\mathbf{x}d')$ since $\sum_{\mathbf{S}-\mathbf{S}'}(\prod_{X\in\mathbf{T}-\mathbf{S}'}\mathbf{P}_X)$ does not involve D, and $\mathbb{C}'(\mathbf{x}d)=\mathbb{C}'(\mathbf{x}d')$, since $\mathbb{C}'$ is a constant function. Thus $\mathbf{P}$ does not depend on D. $\square$

Proposition 2 follows immediately from part (5)(ii) of the following proposition. The proof relies on a long inductive argument, to show that if the five properties hold for Φ^{i+1} then they hold for Φ^i.

Proposition 16. *Let $\mathfrak{U}$ be an uncertainty-utility values structure with operation $+$ on utility values, and let $\vee$ be a disjunctive operation for $\mathfrak{U}$. Let $\mathfrak{I}=\langle G,(\Phi,\Psi)\rangle$ be an ID-system over $\mathfrak{U}$, and let τ be a legal elimination sequence for $\mathfrak{I}$. For $i=0,\ldots,n$, let (Φ^i,Ψ^i) be $\mathbb{M}^{+,\vee}_{\tau[i+1:n]}(\Phi,\Psi)$, where $\tau[n+1:n]$ is defined to be empty sequence, and so $(\Phi^n,\Psi^n)=(\Phi,\Psi)$.*

Consider any $i=0,\ldots,n$. Let k be the unique value such that $\tau[i+1:n]$ does not contain decision variable D_k, but contains decision variable D_l for all $l>k$. Write Φ^i as $\{\mathbf{P}_j : j=1,\ldots,h\}$.

(1) For each $j=1,\ldots,h$, $\mathbf{P}_j$ does not involve decision variable D_l (i.e., $D_l\notin sc(\mathbf{P}_j)$) for any $l\geq k+1$;

(2) $\mathbf{P}_j$ can be written as $\mathbb{C}_j\times\sum_{\mathbf{S}_j}\prod_{X\in\mathbf{T}_j}\mathbf{P}_X$, where $\mathbb{C}_j$ is a constant function, $\mathbf{T}_j\subseteq\mathbf{X}$ and $\mathbf{S}_j=\{Y_{i+1},\ldots,Y_n\}\cap\mathbf{T}_j$.

(3) For $j,g\in\{1,\ldots,h\}$ with $j\neq g$, we have $\mathbf{T}_j\cap\mathbf{T}_g=\emptyset$ and $\mathbf{S}_j\cap sc(\mathbf{P}_g)=\emptyset$.

(4) *For all chance variables $X' \in \{Y_1, \ldots, Y_i\}$, there exists some $\mathbf{P}_j \in \Phi^i$, such that $\mathbf{T}_j$ contains X'.*

(5) *Suppose that D_k is the next variable to be eliminated, i.e., $Y_i = D_k$. Then*

   (i) *$\mathbf{S}_j$ contains all G-descendants of D_k which are in $\mathbf{T}_j$;*

   (ii) *$\mathbf{P}_j$ can be written in such a way that no $\mathbf{P}_X$, for $X \in \mathbf{T}_j$, involves D_k. Thus $\mathbf{P}_j$ does not depend on D_k.*

Proof: We need to show that for each $i = 0, \ldots, n$, Properties (1)–(5) hold for Φ^i. We will prove this by descending induction on i beginning with $i = n$.

Base Case, $i = n$:. We must show that the result holds for n. For this case, $\tau[i+1 : n]$ is the empty sequence of variables, and so does not contain decision variable D_m, so $k = m$ in the conditions. We can label $\mathbf{X}$ as $X_1, \ldots, X_h$, and, for $j = 1, \ldots, h$, let $\mathbf{P}_j = \mathbf{P}_{X_j}$. Then $\Phi^n = \Phi = \{\mathbf{P}_j \,:\, j = 1, \ldots, h\}$. Now, $\mathbf{P}_j$

Consider an arbitrary $\mathbf{P} \in \Phi^i$. Then $\mathbf{P}$ is equal to $\mathbf{P}_X$ for some X.

(1) holds vacuously since there are no decision variables D_l with $l \geq m + 1$.

(2): For $j = 1, \ldots, h$, $\mathbf{P}_j$ can be written as $\mathbb{C} \times \sum_{\mathbf{S}_j} \prod_{X \in \mathbf{T}_j} \mathbf{P}_{X_j}$, where, $\mathbb{C}$ is the constant function with scope $\{X_j\}$ which assigns the value 1 to every value of X_j; $\mathbf{S}_j$ is the empty set, and $\mathbf{T}_j$ is the singleton set $\{X_j\}$.

(3) follows immediately.

(4) also follows immediately, since for all $X' \in \mathbf{X}$ there exists some j with $X_j = X'$, so $\mathbf{T}_j = \{X_j\} \ni X'$.

(5) Suppose that D_k (i.e., D_m) is the next variable to be eliminated. $\mathbf{I}_m = \emptyset$, so Lemma 36(ii) implies that D_k has no descendants. This implies (5)(i), and also that no $\mathbf{P}_X$ involves D_k (since this would imply that D_k would be a parent of X. Thus, (5)(ii) also holds for the base case.

Inductive Case.

Suppose that Properties (1)–(5) hold for Φ^{i+1}. We will show that Properties (1)–(5) hold for Φ^i.

The approach we take is as follows.

(A) We will show that the elements of Φ^i can be written in the form specified in Property (2), and that Properties (1), (3) and (4) hold for this form. There are different arguments according to whether the previously eliminated variable Y_{i+1} is a chance or a decision variable. This then proves the result for the case when $D_k \neq Y_i$, i.e., when the next variable to be eliminated is a chance variable.

(B) We then focus on the case when $Y_i = D_k$, i.e., the next variable to be eliminated is D_k. We use Lemma 37 to show that the representation in condition (2) can be modified so that each $\mathbf{P}_X$, for $X \in \mathbf{T}_j$, does not involve D so Property (5) holds. We also then need to show that Properties (1)–(4) still hold for this modified form.

Let k be the unique value such that $\tau[i+1 : n]$ does not contain decision variable D_k, but contains decision variable D_j for all $j > k$. There are two cases, (a) Y_{i+1} is a chance variable; or (b) Y_{i+1} is a decision variable, so that $Y_{i+1} = D_{k+1}$.

Part (A) (a): chance variable case. Y_{i+1} is a chance variable, which we call X'.

Let $\Phi^{i+1}_{\ni X'}$ be the multi-set of elements $\mathbf{P}$ of Φ^{i+1} with $sc(\mathbf{P}) \ni X'$, and let $\Phi^{i+1}_{\not\ni X'}$ be the other elements of Φ^{i+1}. The collection Φ^i is then equal to $\Phi^{i+1}_{\not\ni X'} \cup \{\mathbf{P}'\}$, where $\mathbf{P}' = \sum_{X'} \mathbf{P}^+$, and $\mathbf{P}^+ = \prod_{\mathbf{P} \in \Phi^{i+1}_{\ni X'}} \mathbf{P}$ (see Section 4.1). Write Φ^{i+1} as $\{\mathbf{P}^l : 1, \ldots, f\}$, in such a way that $\{\mathbf{P}^l : 1, \ldots, g\}$ ($g \leq f$) are the elements of Φ^{i+1} that involve X' (i.e., the elements of $\Phi^{i+1}_{\ni X'}$), and so $\{\mathbf{P}^l : g+1, \ldots, f\} = \Phi^{i+1}_{\not\ni X'}$.

By the inductive hypothesis (Property (1) for Φ^{i+1}), for all $l = 1 \ldots, f$, $sc(\mathbf{P}^l) \cap \{D_{k+1}, \ldots D_m\} = \emptyset$. This implies $sc(\mathbf{P}') \cap \{D_{k+1}, \ldots D_m\} = \emptyset$, and thus for all $\mathbf{P} \in \Phi^i$, $sc(\mathbf{P}) \cap \{D_{k+1}, \ldots D_m\} = \emptyset$. This proves Property (1) for Φ^i for Part (A), Case (a).

By the inductive hypothesis, each $\mathbf{P}^l$ can be written in the form $\mathbb{C}_l \times \sum_{\mathbf{S}_l} \prod_{X \in \mathbf{T}_l} \mathbf{P}_X$, where $\mathbf{S}_l = \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}_l$ (Property (2) for Φ^{i+1}). For each l, we have that $X' \notin \mathbf{S}_l$, since $X' \notin \{Y_{i+2}, \ldots, Y_n\}$. Property (4) for Φ^{i+1} implies that for some l', $\mathbf{T}_{l'} \ni X'$. Since $X' \notin \mathbf{S}_{l'}$, $\mathbf{P}_{l'}$ involves X', and so $l' \leq g$. Then, for $l > g$, $X' \notin \mathbf{T}_l$, since the sets T_l are disjoint (by Property (3) for Φ^{i+1}). This implies that $\mathbf{S}_l = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}_l$ for $l > g$ (since $Y_{i+1} = X' \notin \mathbf{T}_l$). This establishes Property (2) for $\Phi^i - \{\mathbf{P}'\}$.

Using Lemma 35(ii), we can write

$$\mathbf{P}' = \mathbb{C} \times \sum_{X'} \prod_{l=1}^{g} \sum_{\mathbf{S}_l} \prod_{X \in \mathbf{T}_l} \mathbf{P}_X,$$

where $\mathbb{C}$ is a constant function.

Let us write $\mathbf{P}_*^l = \prod_{X \in \mathbf{T}_l} \mathbf{P}_X$. For $l' \neq l$, by property (3) for Φ^{i+1}, we have $\mathbf{S}_l \cap \mathbf{S}_{l'} = \emptyset$ (because $\mathbf{T}_l \cap \mathbf{T}_{l'} = \emptyset$, and $\mathbf{S}_l \subseteq \mathbf{T}_l$ and $\mathbf{S}_{l'} \subseteq \mathbf{T}_{l'}$). We also have $\mathbf{S}_l \cap sc(\mathbf{P}_{l'}) = \emptyset$, and therefore $\mathbf{S}_l \cap sc(\mathbf{P}_*^{l'}) = \emptyset$, since $sc(\mathbf{P}_*^{l'}) \subseteq sc(\mathbf{P}_{l'}) \cup S_{l'}$. We can then apply Lemma 34 to give

$$\prod_{l=1}^{g} \sum_{\mathbf{S}_l} \mathbf{P}_*^l = \sum_{\mathbf{S}} \prod_{l=1}^{g} \mathbf{P}_*^l = \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X,$$

where

$$\mathbf{S} = \bigcup_{l=1}^{g} \mathbf{S}_l \text{ and } \mathbf{T} = \bigcup_{l=1}^{g} \mathbf{T}_l.$$

Each variable in each $\mathbf{S}_l$ is in $\{Y_{i+2}, \ldots, Y_n\}$, so X' is not in $\mathbf{S}$. Thus $\mathbf{P}'$ is equal to $\mathbb{C} \times \sum_{X'} \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$, and so

$$\mathbf{P}' = \mathbb{C} \times \sum_{\{X'\} \cup \mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X.$$

We will next show that $\{X'\} \cup \mathbf{S} = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. As observed earlier, there exists some $l \in \{1, \ldots, g\}$ such that $X' \in \mathbf{T}_l$, so $X' \in \mathbf{T}$. $X' = Y_{i+1}$ so $X' \in \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. For all $l \in \{1, \ldots, g\}$, $\mathbf{S}_l = \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}_l$, thus $\mathbf{S}_l \subseteq \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}$, and so $\mathbf{S} \subseteq \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}$. Therefore, $\{X'\} \cup \mathbf{S} \subseteq \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. Conversely, consider any element $X \in \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. If $X = Y_{i+1}$ then $X = X'$, so $X \in \{X'\} \cup \mathbf{S}$. Otherwise, for some $l \in \{1, \ldots, g\}$, $X \in \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}_l$, so $X \in \mathbf{S}_l$, so $X \in \{X'\} \cup \mathbf{S}$. Thus $\{X'\} \cup \mathbf{S} \supseteq \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$ and hence $\{X'\} \cup \mathbf{S} = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$.

We have thus shown that every $\mathbf{P}_j \in \Phi^i$ can be written as $\mathbb{C}_j \times \sum_{\mathbf{S}_j} \prod_{X \in \mathbf{T}_j} \mathbf{P}_X$, where, $\mathbb{C}_j$ is a constant function, and $\mathbf{S}_j = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. This establishes Property (2) for Part (A) Case (a).

Establishing Property (3): Recall that $\Phi^i = \{\mathbf{P}'\} \cup \{\mathbf{P}^{f+1}, \ldots, \mathbf{P}^g\}$, and that $\mathbf{P}' = \mathbb{C} \times \sum_{\{X'\} \cup \mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$. To show that Property (3) holds for Φ^i we need to show that following, for each $l, l' \in \{g+1, \ldots, f\}$ with $l \neq l$. (I) $\mathbf{T}_l \cap \mathbf{T}_{l'} = \emptyset$;

(II) $\mathbf{T}_l \cap \mathbf{T} = \emptyset$; (III) $\mathbf{S}_l \cap sc(\mathbf{P}^{l'}) = \emptyset$; (IV) $\mathbf{S}_l \cap sc(\mathbf{P}') = \emptyset$; (V) $(\{X'\} \cup \mathbf{S}) \cap sc(\mathbf{P}^l) = \emptyset$. Since $\{\mathbf{P}^{g+1}, \ldots, \mathbf{P}^f\} \subseteq \Phi^{i+1}$, Property (3) for Φ^{i+1} immediately implies (I) and (III). Property (3) for Φ^{i+1} also implies for all $l'' \in \{1, \ldots, g\}$, $\mathbf{T}_{l''} \cap \mathbf{T}_l = \emptyset$, and thus (II), since $\mathbf{T} = \bigcup_{l''=1}^{g} \mathbf{T}_{l''}$. For each $l'' \in \{1, \ldots, g\}$, (i) $\mathbf{S}_l \cap sc(\mathbf{P}^{l''}) = \emptyset$, and (ii) $sc(\mathbf{P}^l) \cap S_{l''} = \emptyset$, again by Property (3) for Φ^{i+1}. Then, $sc(\mathbf{P}') \subseteq \bigcup_{l''=1}^{g} sc(\mathbf{P}^{l''})$, so by (i), (IV) holds. (ii) implies $\mathbf{S} \cap sc(\mathbf{P}^l) = \emptyset$, since $\mathbf{S} = \bigcup_{l=1}^{g} \mathbf{S}_l$. By definition of $\Phi^{i+1}_{\not\ni X'}$, $sc(\mathbf{P}^l) \not\ni X'$. This establishes (V), and hence Property (3) for Φ^i.

Establishing Property (4): Consider any chance variable $X \in \{Y_1, \ldots, Y_i\}$. To establish (4) for Φ^i we need to show that either $\mathbf{T} \ni X$ or, for some $l \in \{g+1, \ldots, f\}$, $\mathbf{T}_l \ni X$. Property (4) for Φ^{i+1} implies that for some $l' \in \{1, \ldots, f\}$, $\mathbf{T}_{l'} \ni X$, so $\bigcup_{l'=1}^{f} \mathbf{T}_{l'} \ni X$. Since $\mathbf{T} = \bigcup_{l=1}^{g} \mathbf{T}_l$, either $\mathbf{T} \ni X$ or for some $l \in \{g+1, \ldots, f\}$, $\mathbf{T}_l \ni X$, and so Property (4) holds for Φ^i.

Part (A) (b): decision variable case. $Y_{i+1} = D_{k+1}$.

Property (5) for Φ^{i+1} implies that no element of Φ^{i+1} depends on D_{k+1}, so, by definition, $\Phi^i = \{\mathbf{P}^{-D_{k+1}} : \mathbf{P} \in \Phi^{i+1}\}$. Consider any $\mathbf{P}^{-D_{k+1}} \in \Phi^i$. Properties (2) and (5) for Φ^{i+1} imply that $\mathbf{P}$ can be written as $\mathbb{C} \times \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$, where c is a constant function, $\mathbf{T} \subseteq \mathbf{X}$ and $\mathbf{S} = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$, and for $X \in \mathbf{T}$, $\mathbf{P}$ does not involve D_{k+1}. Then, $\mathbf{P}^{-D_{k+1}} = \mathbb{C}' \times \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$, where $\mathbb{C}'$ is a constant function with scope $sc(\mathbb{C}) - \{D_{k+1}\}$. Now, $\mathbf{S} = \{Y_{i+2}, \ldots, Y_n\} \cap \mathbf{T}$ since $Y_{i+1} \notin \mathbf{T}$ as it is not a chance variable. This establishes Property (2) for Φ^i for Part (A) Case (b). Properties (3) and (4) also follow, since the sets $\mathbf{S}$ and $\mathbf{T}$ have not changed from the representations for Φ^{i+1} to the representations for Φ^i.

Property (1) for Φ^{i+1} implies that $\mathbf{P}$ does not involve D_l for any $l \geq k+2$; thus $\mathbf{P}^{-D_{k+1}}$ does not involve D_l for any $l \geq k+2$. Also, $\mathbf{P}^{-D_{k+1}}$ does not involve D_{k+1}, so $\mathbf{P}^{-D_{k+1}}$ does not involve D_l for any $l \geq k+1$. This establishes Property (1) for Φ^i.

Part (B) The case when $Y_i = D_k$:

We consider here the case where D_k is the next variable to be eliminated, i.e., $Y_i = D_k$.

We established in Part (A) that any $\mathbf{P} \in \Phi^i$ can be written as $\mathbb{C} \times \sum_{\mathbf{S}} \prod_{X \in \mathbf{T}} \mathbf{P}_X$, where $\mathbb{C}$ is a constant function, and $\mathbf{T} \subseteq \mathbf{X}$ and $\mathbf{S} = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$.

We first show that $\mathbf{S}$ contains all G-descendants of D_k which are in $\mathbf{T}$. Let Y be a G-descendant of D_k which is in $\mathbf{T}$. Y is a chance variable (since it is in $\mathbf{T}$), so Lemma 36(ii) implies that $Y \in \mathbf{I}_k \cup \cdots \cup \mathbf{I}_m$. Now, $\mathbf{I}_k \cup \cdots \cup \mathbf{I}_m \subseteq \{Y_{i+1}, \ldots, Y_n\}$, and $Y \in \mathbf{T}$, so $Y \in \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T} = \mathbf{S}$. This shows that $\mathbf{S}$ contains all G-

descendants of D_k which are in $\mathbf{T}$, which establishes Property (5)(i) for Φ^i.

Then Lemma 37 implies that $\mathbf{P}$ can be written as $\mathbb{C}' \times \sum_{\mathbf{S}-\mathbf{S}'} \prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X$, where $\mathbf{S}'$ is the set of elements of $\mathbf{T}$ that are G-descendants of $\mathbf{T}$, and for all $X \in \mathbf{T}-\mathbf{S}'$, $\mathbf{P}_X$ does not involve D, and so $\mathbf{P}$ does not depend on D. This shows Property (5)(ii) for Φ^i, and Lemma 37 also implies Property 5(i) still holds.

Since we are changing the representation of $\mathbf{P}$, we need to show that Properties (2), (3) and (4) are still satisfied by the new representation. (Property (1) obviously still holds, since it is property of $\mathbf{P}$ rather than of the representation $\mathbb{C}' \times \sum_{\mathbf{S}-\mathbf{S}'} \prod_{X \in \mathbf{T}-\mathbf{S}'} \mathbf{P}_X$.)

Regarding Property (2), we just need to show that $\mathbf{S}-\mathbf{S}' = \{Y_{i+1}, \ldots, Y_n\} \cap (\mathbf{T}-\mathbf{S}')$. We have that $\mathbf{S} = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T}$. Thus $\mathbf{S} \cap (\mathbf{X}-\mathbf{S}') = \{Y_{i+1}, \ldots, Y_n\} \cap \mathbf{T} \cap (\mathbf{X}-\mathbf{S}')$, and thus $\mathbf{S}-\mathbf{S}' = \{Y_{i+1}, \ldots, Y_n\} \cap (\mathbf{T}-\mathbf{S}')$.

Property (3) follows immediately since $\mathbf{T}' \subseteq \mathbf{T}$ and $\mathbf{S}' \subseteq \mathbf{S}$; in the new representation, the sets are being reduced (or staying as they are), so Property (3) is maintained.

To prove that Property (4) still holds (for the new representation) it is sufficient to show that if X' is a chance variable in $\{Y_1, \ldots, Y_i\}$ such that $\mathbf{T} \ni X'$ then $\mathbf{T}-\mathbf{S}' \ni X'$. In other words, if $X' \in \mathbf{T} \cap \{Y_1, \ldots, Y_i\}$, then $X' \notin \mathbf{S}'$. But this follows from the fact that $\mathbf{S}' \subseteq \mathbf{S} \subseteq \{Y_{i+1}, \ldots, Y_n\}$.

We have thus shown in all cases that Properties (1)–(5) hold for Φ^i. This completes the inductive step, so Properties (1)–(5) hold for Φ^i for all $i = 0, \ldots, n$, completing the proof of Proposition 16. $\square$

Part C: Proofs of Results in Sections 7 and 8

Proposition 3. *For any u.u.v. structure $\mathfrak{U}$, the associated tuple $\mathfrak{U}^*$ is a weak uncertainty-utility values structure.*

Proof: Firstly, $\langle Q, +_Q, \times_Q, 0_Q, 1\rangle$ is an uncertainty values structure. Let A, B, C, A_1, A_2 be subsets of U, and let q, q_1, q_2 be elements of Q. Operation $+$ on subsets of U is clearly commutative. $A + \{0_U\} = \{a + 0_U \,:\, a \in A\} = A$, so $\{0_U\}$ is the additive identity. Using associativity of $+$ on U:

$$\begin{aligned}(A+B)+C &= \{a+b \,:\, a \in A, b \in B\} + C \\ &= \{a+b+c \,:\, a \in A, b \in B, c \in C\}\end{aligned}$$

Similarly, $A + (B + C) = \{a + b + c \,:\, a \in A, b \in B, c \in C\}$, proving associativity of $+$. Thus, $\langle 2^U, +, \{0_U\}\rangle$ is a utility values structure. To complete the proof, we verify properties ($*1$), ($*2$) and ($*3$) for $\mathfrak{U}^*$.

(∗1):

$$1 \times A = \{1 \times a \,:\, a \in A\} = A$$

(∗2):

$$\begin{aligned} q_1 \times (q_2 \times A) &= q_1 \times \{q_2 \times a \,:\, a \in A\} \\ &= \{q_1 \times (q_2 \times a) \,:\, a \in A\} \\ &= \{(q_1 \times q_2) \times a \,:\, a \in A\} \\ &= (q_1 \times q_2) \times A \end{aligned}$$

(∗3):

$$\begin{aligned} q \times (A_1 + A_2) &= q \times \{a_1 + a_2 \,:\, a_1 \in A_1, a_2 \in A_2\} \\ &= \{q \times (a_1 + a_2) \,:\, a_1 \in A_1, a_2 \in A_2\} \\ &= \{(q \times a_1) + (q \times a_2) \,:\, a_1 \in A_1, a_2 \in A_2\} \end{aligned}$$

We also have

$$\begin{aligned} (q \times A_1) + (q \times A_2) &= \{q \times a_1 \,:\, a_1 \in A_1\} + \{q \times a_2 \,:\, a_2 \in A_2\} \\ &= \{(q \times a_1) + (q \times a_2) \,:\, a_1 \in A_1, a_2 \in A_2\}, \end{aligned}$$

showing Property (*3) for $\mathfrak{U}^*$. □

Lemma 6. *Let $\mathfrak{U} = \langle Q, +_Q, \times_Q, 0_Q, 1, U, +, 0, \times\rangle$, be an uncertainty-utility values structure. For any subsets A, B, C of U, $(A \cup B) + C = (A + C) \cup (B + C)$, and for any $q \in Q$, $q \times (A \cup B) = (q \times A) \cup (q \times B)$.*

Proof: The definitions immediately imply that $(A + C) \subseteq (A \cup B) + C$ and $(B + C) \subseteq (A \cup B) + C$, so $(A \cup B) + C \supseteq (A + C) \cup (B + C)$. We will next show that $(A \cup B) + C \subseteq (A + C) \cup (B + C)$. Let u be an element of $(A \cup B) + C$, which can therefore be written as $d + c$ where $d \in A \cup B$ and $c \in C$. Either (I) $d \in A$, and so $d + c \in A + C$, or (II) $d \in B$, and so $d + c \in B + C$. In either case, $u \in (A + C) \cup (B + C)$.

For the last part: $q \times (A \cup B) = \{q \times c \,:\, c \in A \cup B\} = \{q \times a \,:\, a \in A\} \cup \{q \times b \,:\, b \in B\} = (q \times A) \cup (q \times B)$. □

Lemma 7. *For any $A \subseteq U$,*

(i) $\mathcal{C}(A)$ is a convex set containing A.

(ii) $\mathcal{C}(A)$ *is equal to the intersection of all convex sets containing* A*, and is therefore the unique smallest convex set containing* A*.*

(iii) A *is convex if and only if* $\mathcal{C}(A) = A$*.*

(iv) $\mathcal{C}(\mathcal{C}(A)) = \mathcal{C}(A)$*.*

(v) If $B \subseteq A$ *then* $\mathcal{C}(B) \subseteq \mathcal{C}(A)$*.*

Proof: (i): Setting $k = 1$ and a_1 to be an arbitrary element of A, we have $1 \times a_1 \in \mathcal{C}(A)$, and hence $a_1 \in \mathcal{C}(A)$. This shows that $\mathcal{C}(A)$ contains A.

We next show that $\mathcal{C}(A)$ is convex. Consider elements $f, g \in \mathcal{C}(A)$, and consider any $r, s \in Q$ with $r + s = 1$. We need to show that $(r \times f) + (s \times g) \in \mathcal{C}(A)$.

Since $f, g \in \mathcal{C}(A)$, we can write f as $\sum_{i=1}^{k}(p_i \times a_i)$ where each $a_i \in A$, each $p_i \in Q$, and $\sum_{i=1}^{k} p_i = 1$, and we can write g as $\sum_{j=1}^{l}(q_j \times b_j)$, where each $b_j \in A$, each $q_i \in Q$ and $\sum_{j=1}^{l} q_j = 1$.

Then $r \times f$ equals to $\sum_{i=1}^{k} r \times (p_i \times a_i)$, using iterative use of $(*3)$ which equals $\sum_{i=1}^{k}((r \times p_i) \times a_i)$, using $(*2)$. Similarly, $s \times g$ equals $\sum_{j=1}^{l}((s \times q_j) \times b_j)$. Hence, $(r \times f) + (s \times g)$ equals $\sum_{i=1}^{k}((r \times p_i) \times a_i) + \sum_{j=1}^{l}((s \times q_j) \times b_j)$, which is in $\mathcal{C}(A)$ given that we also have $\sum_{i=1}^{k}(r \times p_i) + \sum_{j=1}^{l}(s \times q_j) = 1$.

The distributive property of uncertainty values implies that $\sum_{i=1}^{k}(r \times p_i) = r \times \sum_{i=1}^{k} p_i = r \times 1 = r$. Also, $\sum_{j=1}^{l}(s \times q_j) = s \times \sum_{j=1}^{l} q_j = s \times 1 = s$. Hence, $\sum_{i=1}^{k}(r \times p_i) + \sum_{j=1}^{l}(s \times q_j) = r + s = 1$.

(ii): Let B be any convex set containing A. We will show by induction that for any $k \geq 1$, $\sum_{i=1}^{k}(q_i \times a_i) \in B$, when each a_i is in A, each q_i is in Q, and $\sum_{i=1}^{k} q_i = 1$. This proves that $B \supseteq \mathcal{C}(A)$.

Base case: if $k = 1$, then $1 \times a_1 = a_1 \in A$ and so $1 \times a_1 \in B$ since $B \supseteq A$.

Suppose that the inductive hypothesis is true for $k = l$. We will show that it is true for $k = l + 1$. Consider any expression of the form $\sum_{i=1}^{l+1}(q_i \times a_i)$ (where $\sum_{i=1}^{l+1} q_i = 1$) which can be written in the form $(q \times b) + (q_{l+1} \times a_{l+1})$, where $q = \sum_{i=1}^{l} q_i$ and $b = q^{-1} \times \sum_{i=1}^{l}(q_i \times a_i)$. We need to show that $q \times b + q_{l+1} \times a_{l+1} \in B$. Now, $q + q_{l+1} = \sum_{i=1}^{l+1} q_i = 1$. Since B is convex it is sufficient to show that $b, a_{l+1} \in B$. $a_{l+1} \in B$ since $a_{l+1} \in A$ and B contains A. b is equal to $\sum_{i=1}^{l}((q^{-1} \times q_i) \times a_i)$, using $(*2)$ and $(*3)$. We have $\sum_{i=1}^{l}(q^{-1} \times q_i) = q^{-1} \times \sum_{i=1}^{l} q_i$ which equals $q^{-1} \times q = 1$. By the inductive hypothesis, $b \in B$, completing the proof by induction that $B \supseteq \mathcal{C}(A)$. This implies that the intersection of all convex sets containing A contains $\mathcal{C}(A)$, which since, $\mathcal{C}(A)$ is a convex set containing A, implies that $\mathcal{C}(A)$ is equal to the intersection of all convex sets containing A.

(iii): First suppose that A is convex. By (ii), $\mathcal{C}(A)$ is the unique smallest convex set containing A, which equals A since A is convex.

Conversely, if $\mathcal{C}(A) = A$ then A is convex by (i).

(iv): By (i), $\mathcal{C}(A)$ is convex, and hence by (iii), $\mathcal{C}(\mathcal{C}(A)) = \mathcal{C}(A)$.

(v): This follows immediately from the definition of convex closure. □

Proposition 5. *Let A and B be subsets of U, and let q be an element of Q.*

(i) If A and B are convex then $A + B$ is convex.

(ii) $\mathcal{C}(A + B) = \mathcal{C}(A) + \mathcal{C}(B)$.

(iii) $\mathcal{C}(\mathcal{C}(A) + B) = \mathcal{C}(A + B)$.

(iv) $\mathcal{C}(\mathcal{C}(A) \cup B) = \mathcal{C}(A \cup B)$.

(v) $\mathcal{C}(q \times A) = q \times \mathcal{C}(A)$.

(vi) $\mathcal{C}(A \cup B) = \mathcal{C}(A) \cup \mathcal{C}(B) \cup (\mathcal{C}(A) \oplus \mathcal{C}(B))$.

Proof: (i): Consider any pair of elements $a_1 + b_1$ and $a_2 + b_2$ in $A + B$, and any $q_1, q_2 \in Q$ such that $q_1 + q_2 = 1$. It is sufficient to show that $q_1 \times (a_1 + b_1) + q_2 \times (a_2 + b_2) \in A + B$.

Write $a_3 = (q_1 \times a_1) + (q_2 \times a_2)$, and $b_3 = (q_1 \times b_1) + (q_2 \times b_2)$. Since A and B are convex, $a_3 \in A$ and $b_3 \in B$.

$q_1 \times (a_1 + b_1) + q_2 \times (a_2 + b_2) = (q_1 \times a_1) + (q_2 \times a_2) + (q_1 \times b_1) + (q_2 \times b_2) = a_3 + b_3 \in A + B$.

(ii): $A+B \subseteq \mathcal{C}(A)+\mathcal{C}(B)$, and so, by Lemma 7(v), $\mathcal{C}(A+B) \subseteq \mathcal{C}(\mathcal{C}(A)+\mathcal{C}(B))$, which equals $\mathcal{C}(A) + \mathcal{C}(B)$ by part (i) (and Lemma 7(iii)).

We need just to show that $\mathcal{C}(A)+\mathcal{C}(B) \subseteq \mathcal{C}(A+B)$. Consider any $f \in \mathcal{C}(A)$ and $g \in \mathcal{C}(B)$. Write f as $\sum_{i=1}^{k}(p_i \times a_i)$ where each $a_i \in A$, each $p_i \in Q$, and $\sum_{i=1}^{k} p_i = 1$, and write g as $\sum_{j=1}^{l}(q_j \times b_j)$, where each $b_j \in B$, each $q_i \in Q$ and $\sum_{j=1}^{l} q_j = 1$.

Now, $\sum_{i,j}(p_i \times q_j) = (\sum_i p_i) \times (\sum_j q_j) = 1 \times 1 = 1$. Thus, $h = \sum_{i,j}((p_i \times q_j) \times (a_i + b_j)) \in \mathcal{C}(A+B)$. We can write h as $\sum_{i,j}((p_i \times q_j) \times a_i) + \sum_{i,j}((p_i \times q_j) \times b_j)$. $\sum_{i,j}((p_i \times q_j) \times a_i) = \sum_i (p_i \times (\sum_j q_j) \times a_i) = \sum_i (p_i \times a_i) = f$. Similarly,

$\sum_{i,j}((p_i \times q_j) \times b_j) = g$, showing that $h = f + g$, and so $f + g \in \mathcal{C}(A + B)$, completing the proof of (ii).

(iii): By part (ii), $\mathcal{C}(\mathcal{C}(A) + B) = \mathcal{C}(\mathcal{C}(A)) + \mathcal{C}(B)$, which equals $\mathcal{C}(A) + \mathcal{C}(B)$ by Lemma 7(iv). Using part (ii) again gives the result.

(iv): $\mathcal{C}(A \cup B) \supseteq \mathcal{C}(A)$ by Lemma 7(v), and $\mathcal{C}(A \cup B) \supseteq B$ by Lemma 7(i), so $\mathcal{C}(A) \cup B \subseteq \mathcal{C}(A \cup B)$. Applying Lemma 7(v) and (iv) gives $\mathcal{C}(\mathcal{C}(A) \cup B) \subseteq \mathcal{C}(\mathcal{C}(A \cup B)) = \mathcal{C}(A \cup B)$.

Conversely, Lemma 7(v) and (i) imply $\mathcal{C}(\mathcal{C}(A) \cup B) \supseteq \mathcal{C}(A \cup B)$.

(v): Consider any element f of $\mathcal{C}(q \times A)$. f can be written as $\sum_{i=1}^{k} p_i \times (q \times a_i)$, for some elements a_i of A and $p_i \in Q$ such that $\sum_{i=1}^{k} p_i = 1$. $p_i \times (q \times a_i) = (p_i \times q) \times a_i = q \times (p_i \times a_i)$, and so $f = q \times (\sum_{i=1}^{k} p_i \times a_i)$, which implies that $f \in q \times \mathcal{C}(A)$.

Conversely, consider any element f of $q \times \mathcal{C}(A)$. Then f is equal to $q \times g$ for some element g of $\mathcal{C}(A)$. g can be written as $\sum_{i=1}^{k} p_i \times a_i$ for some elements a_i of A and $p_i \in Q$ such that $\sum_{i=1}^{k} p_i = 1$. Then, by the same argument as above, f is equal to $\sum_{i=1}^{k} p_i \times (q \times a_i)$, where $q \times a_i \in q \times A$, and hence $f \in \mathcal{C}(q \times A)$.

(vi): Suppose first that $f \in \mathcal{C}(A \cup B)$. We will show that f is either (I) an element of $\mathcal{C}(A)$, or (II) an element of $\mathcal{C}(B)$, or (III) an element of $\mathcal{C}(A) \oplus \mathcal{C}(B)$.

We can write f as $\sum_{i=1}^{k}(p_i \times a_i) + \sum_{j=1}^{l}(q_j \times b_j)$, where each $a_i \in A$, each $b_j \in B$ and $\sum_{i=1}^{k} p_i + \sum_{j=1}^{l} q_j = 1$. If $k = 0$ or $l = 0$ then f is either an element of $\mathcal{C}(A)$, or an element of $\mathcal{C}(B)$, so let us assume that $k, l \neq 0$.

Let $p = \sum_{i=1}^{k} p_i$ and define, for each i, $p'_i = p^{-1} \times p_i$, and define $a = \sum_{i=1}^{k}(p'_i \times a_i)$. We have: $\sum_{i=1}^{k} p'_i = p^{-1} \times \sum_{i=1}^{k} p_i = 1$, and so $a \in \mathcal{C}(A)$. Similarly, let $q = \sum_{j=1}^{l} q_j$ and define, for each j, $q'_j = q^{-1} \times q_j$, and define $b = \sum_{j=1}^{l}(q'_j \times b_j)$. We have $\sum_{j=1}^{l} q'_j = 1$, and so $b \in \mathcal{C}(B)$. It can be easily seen that $f = (p \times a) + (q \times b)$ and $p + q = 1$, so $f \in \mathcal{C}(A) \oplus \mathcal{C}(B)$.

Conversely, clearly, by monotonicity (Lemma 7(v)), $\mathcal{C}(A) \subseteq \mathcal{C}(A \cup B)$ and $\mathcal{C}(B) \subseteq \mathcal{C}(A \cup B)$. It remains to show that $\mathcal{C}(A) \oplus \mathcal{C}(B) \subseteq \mathcal{C}(A \cup B)$. Consider any $h \in \mathcal{C}(A) \oplus \mathcal{C}(B)$, which can be written as $(p \times f) + (q \times g)$ for some $f \in \mathcal{C}(A)$, $g \in \mathcal{C}(B)$ and $p, q \in Q$ with $p + q = 1$. Write f as $\sum_{i=1}^{k}(p_i \times a_i)$ where each $a_i \in A$, each $p_i \in Q$, and $\sum_{i=1}^{k} p_i = 1$, and write g as $\sum_{j=1}^{l}(q_j \times b_j)$, where each $b_j \in B$, each $q_i \in Q$ and $\sum_{j=1}^{l} q_j = 1$. Then $h = \sum_{i=1}^{k}((p \times p_i) \times a_i) + \sum_{j=1}^{l}((q \times q_j) \times b_j)$. Now, $\sum_{i=1}^{k}(p \times p_i) + \sum_{j=1}^{l}(q \times q_j) = (p \times \sum_{i=1}^{k} p_i) + (q \times \sum_{j=1}^{l} q_j)$, which equals $p + q = 1$, and so $h \in \mathcal{C}(A \cup B)$, as required. $\square$

Lemma 9. *Let A be a subset of U.*

(i) $\max_{\succeq}(A) \subseteq \mathcal{R}_{\succeq}(A)$

(ii) If A *satisfies* MAX *then* $\max_{\succeq}(A) = \mathcal{R}_{\succeq}(A)$.

Proof: (i) Suppose $a \in \max_{\succeq}(A)$. Then there does not exist $b \in \max_{\succeq}(A)$ with $b \succ a$ and so $a \in \mathcal{R}_{\succeq}(A)$.

(ii) Suppose that A satisfies MAX. By (i), it is sufficient to prove that $\mathcal{R}_{\succeq}(A) \subseteq \max_{\succeq}(A)$. Suppose $a \notin \max_{\succeq}(A)$. Then, by MAX, there exists $b \in \max_{\succeq}(A)$ with $b \succ a$, which implies that $b \notin \mathcal{R}_{\succeq}(A)$. $\square$

Lemma 10. *Let* A *and* B *be subsets of* U. *Then* $\max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B) = \max_{\succeq}(A \cup B)$.

Proof: Suppose $c \in \max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B) - \max_{\succeq}(A \cup B)$. Then $c \in \mathcal{R}_{\succeq}(A) \cup B$ so $c \in A \cup B$. Since $c \notin \max_{\succeq}(A \cup B)$ there exists $d \in A \cup B$ with $d \succ c$. Since $c \in \max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B)$, $d \notin \mathcal{R}_{\succeq}(A) \cup B$, so $d \notin \mathcal{R}_{\succeq}(A)$. Then there exists $e \in \max_{\succeq}(A)$ with $e \succ d$, and so $e \succ c$. Also, $e \in \mathcal{R}_{\succeq}(A)$ and so $e \in \mathcal{R}_{\succeq}(A) \cup B$, which contradicts $c \in \max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B)$.

Conversely, suppose $c \in \max_{\succeq}(A \cup B)$. Then c is not dominated by any other element of $A \cup B$, and so $c \in \mathcal{R}_{\succeq}(A) \cup B$, and hence $c \in \max_{\succeq}(\mathcal{R}_{\succeq}(A) \cup B)$ (since $\mathcal{R}_{\succeq}(A) \subseteq A$). $\square$

Lemma 11. *Let* A *and* B *be subsets of* U. *Then,* $A \approx B$ *implies* $\max_{\succeq}(A) = \max_{\succeq}(B)$. *Furthermore, if* A *satisfies* MAX *then the converse also holds, so we have* $A \approx B$ *if and only if* $\max_{\succeq}(A) = \max_{\succeq}(B)$.

Proof: Assume that $A \approx B$. We will prove that $\max_{\succeq}(A) \subseteq \max_{\succeq}(B)$. Reversing the roles of A and B then gives $\max_{\succeq}(B) \subseteq \max_{\succeq}(A)$, and hence $\max_{\succeq}(A) = \max_{\succeq}(B)$.

Suppose $a \in \max_{\succeq}(A)$. Since $A \preccurlyeq B$ there exists $b \in B$ with $a \preceq b$. Either (I) $b \in \max_{\succeq}(B)$ and we define $b'' = b$, or (II) there exists $b' \in B$ with $b' \succ b$ and we define $b'' = b'$. Since $A \succcurlyeq B$ there exists $a' \in A$ with $a' \succeq b''$, and so $a' \succeq b'' \succeq b \succeq a$ and so $a' \succeq a$. Since $a \in \max_{\succeq}(A)$ this implies that $a' = a$ and so $b'' = b = a$ (since $\succeq$ is a partial order). If (II) then $b \succ b''$ which is a contradiction. Hence $b \in \max_{\succeq}(B)$, proving $\max_{\succeq}(A) \subseteq \max_{\succeq}(B)$.

Suppose now that A satisfies MAX and $\max_{\succeq}(A) = \max_{\succeq}(B)$. Consider any $a \in A$. Since A satisfies MAX there exists $c \in \max_{\succeq}(A)$ with $a \preceq b$. But then $b \in \max_{\succeq}(B)$, so, in particular, $b \in B$. Hence for all $a \in A$ there exists $b \in B$

with $b \succeq a$, so $A \preccurlyeq B$. By the same argument with A and B reversed, we have $B \preccurlyeq A$, and hence, $A \approx B$, as required. $\square$

Lemma 12. *Let A, B and C be subsets of U, and let q be an element of Q. Suppose that $A \approx B$. Then*

(i) $q \times A \approx q \times B$*;*

(ii) $A + C \approx B + C$*;*

Proof: (i) We need to show that $q \times A \approx q \times B$. We will show that $q \times A \preccurlyeq q \times B$; the same argument, reversing the roles of A and B, will then imply that $q \times A \succcurlyeq q \times B$, proving $q \times A \approx q \times B$.

Any element of $q \times A$ can be written as $q \times a$ for some $a \in A$. Since $A \approx B$ (and hence $A \preccurlyeq B$), there exists some element $b \in B$ with $a \preceq b$. Since $\preceq$ respects $\times$, $q \times a \preceq q \times b$, showing that $q \times A \preccurlyeq q \times B$.

(ii) Again it is sufficient to show that $A + C \preccurlyeq B + C$, since $A + C \succcurlyeq B + C$ follows by the symmetric argument. Consider any $a + c \in A + C$. Since $A \preccurlyeq B$, there exists $b \in B$ with $a \preceq b$, and hence, because $\succeq$ respects $+$, $a + c \preceq b + c$, showing that $A + C \preccurlyeq B + C$. $\square$

Proposition 6. *For any subset A of U, $A \equiv \mathcal{R}_{\succeq}(A)$ and $A \equiv \mathcal{C}(A)$. If A satisfies* MAX *(in particular, if A is finite) then $A \equiv \max_{\succeq}(A)$.*

Proof: $A \equiv \mathcal{C}(A)$ follows immediately, since, using Lemma 7(iv), $\mathcal{C}(A) = \mathcal{C}(\mathcal{C}(A))$ and so $\mathcal{C}(A) \approx \mathcal{C}(\mathcal{C}(A))$, implying $A \equiv \mathcal{C}(A)$.

We will next show that $A \equiv \mathcal{R}_{\succeq}(A)$, i.e., that $\mathcal{C}(A) \succcurlyeq \mathcal{C}(\mathcal{R}_{\succeq}(A))$ and $\mathcal{C}(A) \preccurlyeq \mathcal{C}(\mathcal{R}_{\succeq}(A))$. $\mathcal{C}(A) \succcurlyeq \mathcal{C}(\mathcal{R}_{\succeq}(A))$ follows immediately since $A \supseteq \mathcal{R}_{\succeq}(A)$ and so $\mathcal{C}(A) \supseteq \mathcal{C}(\mathcal{R}_{\succeq}(A))$, using Lemma 7(v).

We will show $\mathcal{C}(A) \preccurlyeq \mathcal{C}(\mathcal{R}_{\succeq}(A))$. Consider an arbitrary element of $\mathcal{C}(A)$, which we can write as $\sum_{i=1}^{k}(p_i \times a_i)$, where each $a_i \in A$, each $p_i \in Q$, and $\sum_{i=1}^{k} p_i = 1$. For each a_i there exists $a_i' \in \mathcal{R}_{\succeq}(A)$ with $a_i' \succeq a_i$. Define $f' = \sum_{i=1}^{k}(p_i \times a_i')$, which is an element of $\mathcal{C}(\mathcal{R}_{\succeq}(A))$. Since $\succeq$ respects $+$ and $\times$, we have $f' \succeq f$. This shows that $\mathcal{C}(A) \preccurlyeq \mathcal{C}(\mathcal{R}_{\succeq}(A))$, as required.

The last part follows immediately from the first part using Lemma 9(ii). $\square$

Proposition 8. *Let A, B and C be subsets of U, and let q be an element of Q. Suppose that $A \equiv B$. Then* *(i)* $q \times A \equiv q \times B$*;*

(ii) $A + C \equiv B + C$*;*

(iii) $A \cup C \equiv B \cup C$*.*

Proof: (i) By Proposition 5(v), $\mathcal{C}(q \times A) = q \times \mathcal{C}(A)$. Since $A \equiv B$, $\mathcal{C}(A) \approx \mathcal{C}(B)$, and so, by Lemma 12(i), $q \times \mathcal{C}(A) \approx q \times \mathcal{C}(B)$, which equals $\mathcal{C}(q \times B)$. Putting this together, $\mathcal{C}(q \times A) \approx \mathcal{C}(q \times B)$, i.e., $q \times A \equiv q \times B$.

(ii) By, Proposition 5(ii), $\mathcal{C}(A + C) = \mathcal{C}(A) + \mathcal{C}(C)$. We are assuming $A \equiv B$, i.e., $\mathcal{C}(A) \approx \mathcal{C}(B)$, and so, by Lemma 12(ii), $\mathcal{C}(A) + \mathcal{C}(C) \approx \mathcal{C}(B) + \mathcal{C}(C)$, which equals $\mathcal{C}(B + C)$, completing the proof that $\mathcal{C}(A + C) \approx \mathcal{C}(B + C)$, i.e., $A + C \equiv B + C$.

(iii) We need to prove that $\mathcal{C}(A \cup C) \approx \mathcal{C}(B \cup C)$. As for (i) and (ii), it is sufficient to prove that $\mathcal{C}(A \cup C) \preccurlyeq \mathcal{C}(B \cup C)$.

Consider any element f of $\mathcal{C}(A \cup C)$. We need to show that there exists some $g \in \mathcal{C}(B \cup C)$ such that $f \preceq g$. By Proposition 5(vi), f is either (I) an element of $\mathcal{C}(A)$, or (II) an element of $\mathcal{C}(C)$, or (III) is the convex combination of an element of $\mathcal{C}(A)$ and an element of $\mathcal{C}(C)$, i.e., $f = (p \times a) + (q \times c)$ for some $a \in \mathcal{C}(A)$, $c \in \mathcal{C}(C)$ and some $p, q \in Q$ such that $p + q = 1$.

If $f \in \mathcal{C}(A)$ then, since $A \equiv B$, and so $\mathcal{C}(A) \approx \mathcal{C}(B)$, there exists some $b \in \mathcal{C}(B)$ with $f \preceq b$. Also $b \in \mathcal{C}(B \cup C)$, so we can set $g = b$. If $f \in \mathcal{C}(C)$ then $f \in \mathcal{C}(B \cup C)$ so we can set $g = f$.

Consider now case (III) where $f = (p \times a) + (q \times c)$ for some $a \in \mathcal{C}(A)$, $c \in \mathcal{C}(C)$ and some $p, q \in Q$ such that $p + q = 1$. Since $A \equiv B$, there exists some $b \in \mathcal{C}(B)$ with $a \preceq b$.

Since $\preceq$ respects $\times$ and $+$, we have $p \times a \preceq p \times b$ and $f = (p \times a) + (q \times c) \preceq g$ where $g = (p \times b) + (q \times c)$. Clearly, $g \in \mathcal{C}(B \cup C)$, completing the proof. $\square$

Proposition 11. *Let $\mathfrak{I} = \langle G, (\Phi, \Psi) \rangle$ be an $\mathfrak{U}$-ID-system, and let τ be a legal elimination sequence for $\mathfrak{I}$. Then $\mathbb{M}_{\tau}^{+,\cup}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$ is equal to $\{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi} \,:\, \text{policies } \pi\}$ which is the set of all possible values of expected utility over all policies for $\mathfrak{I}$, i.e., $\{EU_{\pi} \,:\, \text{policies } \pi\}$.*

Proof: We write $\mathbb{M}_{\tau}^{+,\cup}(\cdot)$ as $\mathbb{M}_{Y_1}(\mathbb{M}_{Y_2}(\cdots(\mathbb{M}_{Y_n}(\cdot))\cdots))$. Let $\Theta_n = \bigotimes\big(\Phi, \Psi^*\big)$. Θ_n is a $\mathfrak{U}^*$-utility function with scope $\mathbf{X} \cup \mathbf{D} = \{Y_1, \ldots, Y_n\}$. For $j = 1, \ldots, n$, let $\Theta_{j-1} = \mathbb{M}_{Y_j}(\mathbb{M}_{Y_{j+1}}(\cdots(\mathbb{M}_{Y_n}(\Theta_n))\cdots))$. Thus, for $j = 1, \ldots, n$, $\Theta_{j-1} =$

$\mathbb{M}_{Y_j}(\Theta_j)$, and Θ_{j-1} is a $\mathfrak{U}^*$-utility function with scope $V_{j-1} = \{Y_1, \ldots, Y_{j-1}\}$. In particular, $\Theta_0 = \mathbb{M}_{\tau}^{+,\cup}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$.

Consider any policy $(\pi_1, \ldots, \pi_m)$ where π_i is a function from $\Omega(\mathbf{S}_i)$ to $\Omega(D_i)$. We will define an amended marginalization operator $\mathbb{M}^{\pi}_{Y_j}(\cdot)$ for each $j = 1, \ldots, n$. If Y_j is a chance variable, i.e., $Y_j \in \mathbf{X}$, then we define $\mathbb{M}^{\pi}_{Y_j}(\cdot)$ to be the same as $\mathbb{M}_{Y_j}(\cdot)$, i.e., summation over the values of the domain of Y_j.

Otherwise, Y_j is a decision variable: $Y_j \in \mathbf{D}$. In this case, $\mathbb{M}_{Y_j}$ involves a union operation. For $\mathbb{M}^{\pi}_{Y_j}$ we replace the union operation, instead choosing a single set in the union, using the policy π. We describe this in more detail below. Since $Y_j \in \mathbf{D}$, for some $i = 1, \ldots, m$, $Y_j = D_i$. Let Θ be a $\mathfrak{U}^*$-utility function with scope $V_j = \{Y_1, \ldots, Y_j\}$. Define $\mathbb{M}^{\pi}_{Y_j}(\Theta)(\mathbf{y}) = \Theta(\mathbf{y}d')$, where $d' \in \Omega_{D_i}$ is defined as follows: $d' = \pi_i(\mathbf{y}')$, and $\mathbf{y}'$ is tuple $\mathbf{y}$ restricted to $\mathbf{S}_i$ ($= V_j \cap \mathbf{X}$). Recall that $\mathbb{M}_{Y_j}(\Theta)(\mathbf{y}) = \bigcup_{d \in \Omega_{Y_j}} \Theta(\mathbf{y}d)$. In contrast, $\mathbb{M}^{\pi}_{Y_j}(\Theta)(\mathbf{y})$ is $\Theta(\mathbf{y}d)$ for a particular d, rather than being the union over all the d in Ω_{D_i}.

For $j = 1, \ldots, n$, let $\Theta^{\pi}_{j-1} = \mathbb{M}^{\pi}_{Y_j}(\mathbb{M}^{\pi}_{Y_{j+1}}(\cdots(\mathbb{M}^{\pi}_{Y_n}(\Theta_n))\cdots)$, and so $\Theta^{\pi}_{j-1} = \mathbb{M}^{\pi}_{Y_j}(\Theta^{\pi}_j)$. The values of the function Θ_n are singleton sets. It is clear, by an obvious downward induction on j, that the values of Θ^{π}_j are all singleton sets (the non-singleton sets in Θ_j were generated by application of the union operation, which is not used in Θ^{π}_j). Θ^{π}_0 is a constant (a function of no variables) so is a singleton set.

In Lemma 38 below, we give the following properties of Θ^{π}_0: (1) it is equal to $\{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi}\}$; (2) for any policy π, $\Theta^{\pi}_0 \subseteq \Theta_0$; (3) if $a \in \Theta_0$ then there exists some policy π with $\Theta^{\pi}_0 = \{a\}$. Properties (2) and (3) together imply $\Theta_0 = \bigcup\{\Theta^{\pi}_0 : \text{policies } \pi\}$, with (2) implying that the right-hand-side is a subset of the left-hand-side, and with (3) implying the converse. Together with (1) we then have $\Theta_0 = \{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi} : \text{policies } \pi\}$, i.e., $\mathbb{M}_{\tau}^{+,\cup}\big(\bigotimes\big(\Phi, \Psi^*\big)\big)$ is equal to the set $\{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi} : \text{policies } \pi\}$, which equals $\{EU_{\pi} : \text{policies } \pi\}$ (see Section 5), thus proving the result. □

Lemma 38. *With the notation as defined above, we have the following properties.*

(1) $\Theta^{\pi}_0 = \{\sum_{\mathbf{X}}[\bigotimes(\Phi, \Psi)]_{\pi}\}$.

(2) For any policy π, $\Theta^{\pi}_0 \subseteq \Theta_0$.

(3) If $a \in \Theta_0$ *then there exists some policy* π *with* $\Theta^{\pi}_0 = \{a\}$.

Proof: We first prove (1). We have $\Theta^{\pi}_0 = \mathbb{M}^{\pi}_{Y_1}(\cdots(\mathbb{M}^{\pi}_{Y_n}(\Theta_n)\cdots)$. It can be seen that for decision variable Y_j, moving $\mathbb{M}^{\pi}_{Y_j}$ later in the sequence (and so earlier in the application order), will make no difference to the result. Iterating this will lead to all the decision variables being eliminated first (so, being at the end of the sequence), and all the chance variables being eliminated last (and thus being at the beginning of the sequence). Relabel the chance variables in the sequence as $Z_1, \dots, Z_k$, and relabel the decision variables as $W_1, \dots, W_l$. This implies that Θ_0^π can be written as $\mathbb{M}^\pi_{Z_1}(\cdots(\mathbb{M}^\pi_{Z_k}(\mathbb{M}^\pi_{W_1}(\cdots(\mathbb{M}^\pi_{W_l}(\Theta_n)\cdots)$. Now, $\mathbb{M}^\pi_{W_1}(\cdots(\mathbb{M}^\pi_{W_l}(\Theta_n)\cdots)$ involves just instantiating the decision variables with policy π, and so $\mathbb{M}^\pi_{W_1}(\cdots(\mathbb{M}^\pi_{W_l}(\Theta_n)\cdots)$ is just $\{[\bigotimes(\Phi,\Psi)]_\pi\}$. Also, $\mathbb{M}^\pi_{Z_1}\cdots\mathbb{M}^\pi_{Z_k}$ is just $\sum_{\mathbf{X}}$, i.e., summing out all the chance variables. Therefore, $\Theta_0^\pi = \{\sum_{\mathbf{X}}[\bigotimes(\Phi,\Psi)]_\pi\}$.

We next prove (2). There is an obvious monotonicity for addition of sets: if for all $k \in K$, $A_k \subseteq B_k$, then $\sum_{k\in K} A_k \subseteq \sum_{k\in K} B_k$. It follows from this using induction that for all $j = 0, \dots, n$, $\Theta_j^\pi \subseteq \Theta_j$. In particular, $\Theta_0^\pi \subseteq \Theta_0$.

We next prove (3), i.e., that if $a \in \Theta_0$ then there exists some policy π with $\Theta_0^\pi = \{a\}$. Recall, that for $j = 1, \dots, n$, $\Theta_{j-1} = \mathbb{M}_{Y_j}(\Theta_j)$. Thus for any $\mathbf{y} \in \Omega_{V_{j-1}}$, if Y_j is a chance variable, then $\Theta_{j-1}(\mathbf{y}) = \sum_{x\in\Omega_{Y_j}} \Theta_j(\mathbf{y}x)$; and if Y_j is a decision variable, then $\Theta_{j-1}(\mathbf{y}) = \bigcup_{d\in\Omega_{Y_j}} \Theta_j(\mathbf{y}d)$.

We will define a collection Q of partial tuples, with each $\mathbf{y} \in Q$, having an associated utility value $a_\mathbf{y}$. Set Q will be the disjoint union of sets $Q_0, \dots, Q_n$, where, for $j = 0, \dots, n$, Q_j is a subset of Ω_{V_j}, where $V_j = \{Y_1, \dots, Y_j\}$. Specifically, Q_0 just consists just of the assignment $\Diamond$ to the empty set of variables: $Q_0 = \{\Diamond\}$. We also define $a_\Diamond = a$. Note that $\Theta_0(\Diamond)$ just means the same as Θ_0, so we then have $a_\Diamond \in \Theta_0(\Diamond)$.

We now proceed inductively to define $Q_1, \dots, Q_n$. Suppose that, for some $j = 1, \dots, n$, we have defined Q_{j-1}, and for each $\mathbf{y} \in Q_{j-1}$ we have an associated utility value $a_\mathbf{y} \in \Theta_{j-1}(\mathbf{y})$. We will define Q_j, which is a subset of Ω_{V_j}, along with their associated utility values. There are two cases, according to whether the current variable Y_j is a chance or decision variable.

Case (I). Y_j is a chance variable. Define Q_j to be $\{\mathbf{y}x : \mathbf{y} \in Q_{j-1}, x \in \Omega_{Y_j}\}$. (Thus, in particular, $Q_1 = \Omega_{Y_1}$.) Consider any $\mathbf{y} \in Q_{j-1}$. Then, since Y_j is a chance variable, $\Theta_{j-1}(\mathbf{y}) = \sum_{x\in\Omega_{Y_j}} \Theta_j(\mathbf{y}x)$. Since $a_\mathbf{y} \in \Theta_{j-1}(\mathbf{y})$, for each $x \in \Omega_{Y_j}$ there exists some $a_{\mathbf{y}x} \in \Theta_j(\mathbf{y}x)$ such that $a_\mathbf{y} = \sum_{x\in\Omega_{Y_j}} a_{\mathbf{y}x}$. This defines (for Case (I)) $a_\mathbf{z}$ for each $\mathbf{z} \in Q_j$.

Case (II). Y_j is a decision variable D_i. Consider any $\mathbf{y} \in Q_{j-1}$. Then, $\Theta_{j-1}(\mathbf{y}) = \bigcup_{d\in\Omega_{D_i}} \Theta_j(\mathbf{y}d)$, so $a_\mathbf{y} \in \Theta_{j-1}(\mathbf{y})$ implies that there exists some $d_\mathbf{y} \in \Omega_{D_i}$ such that $a_{\mathbf{y}} \in \Theta_j(\mathbf{y}d_{\mathbf{y}})$. Define Q_j to be $\{\mathbf{y}d_{\mathbf{y}} : \mathbf{y} \in Q_{j-1}\}$ (so, in particular, $Q_1 = \{d_\Diamond\}$), and define $a_{\mathbf{y}d_{\mathbf{y}}} = a_{\mathbf{y}}$, which is an element of $\Theta_j(\mathbf{y}d_{\mathbf{y}})$.

In both Cases (I) and (II), for $\mathbf{z} \in Q_j$, we have $a_{\mathbf{z}} \in \Theta_j(\mathbf{z})$. This iterative process defines the whole of Q, where $Q = \bigcup_{j=0}^{n} Q_j$, and, for each $j = 0, \ldots, n$, also defines $a_{\mathbf{y}} \in \Theta_j(\mathbf{y})$ for each $\mathbf{y} \in Q_j$.

We will define a policy π (see Section 5). Recall that π is a sequence of functions $(\pi_1, \ldots, \pi_m)$, where π_i is a function from $\Omega(\mathbf{S}_i)$ to $\Omega(D_i)$. We will define each function π_i in turn. Let $\mathbf{x}$ be any assignment to variables $\mathbf{S}_i$, where $\mathbf{S}_i$ equals $V_{j-1} \cap \mathbf{X}$. There exists a unique $\mathbf{y} \in \Omega_{Y_{j-1}}$ which is in Q_{j-1} and extends $\mathbf{x}$. This is because the definition in Case (II) determines a unique value of each decision variable given the values of the earlier variables, and the definition in Case (I) involves extending tuples with all values of chance variables. We then define $\pi_i(\mathbf{x}) = d_{\mathbf{y}}$, where $d_{\mathbf{y}}$ is defined in Case (II) above. This defines the policy π.

We will prove by downward induction that for all $j = 0, \ldots, n$ and for all $\mathbf{y} \in Q_{j-1}$, we have $\Theta^{\pi}_{j-1}(\mathbf{y}) = \{a_{\mathbf{y}}\}$. The base case is when $j = n$. Consider any $\mathbf{y} \in Q_n$. We have $a_{\mathbf{y}} \in \Theta_n(\mathbf{y})$. Since $\Theta_n(\mathbf{y})$ is a singleton set, it's equal to $\{a_{\mathbf{y}}\}$. $\Theta^{\pi}_n = \Theta_n$, so $\Theta^{\pi}_n(\mathbf{y}) = \{a_{\mathbf{y}}\}$.

Now, we proceed with the inductive part of the proof. Suppose that for some $j \in \{1, \ldots, n\}$ we have $\Theta^{\pi}_j(\mathbf{y}) = \{a_{\mathbf{y}}\}$ for all $\mathbf{y} \in Q_j$. We will prove that $\Theta^{\pi}_{j-1}(\mathbf{y}) = \{a_{\mathbf{y}}\}$ holds for all $\mathbf{y} \in Q_{j-1}$. Recall that $\Theta^{\pi}_{j-1} = \mathbb{M}^{\pi}_{Y_j}(\Theta^{\pi}_j)$. Again there are the two cases.

Case (I). Y_j is a chance variable. By definition of Θ^{π}_{j-1}, we have that $\Theta^{\pi}_{j-1}(\mathbf{y})$ equals $\sum_{x \in \Omega_{Y_j}} \Theta^{\pi}_j(\mathbf{y}x)$. Since $\mathbf{y} \in Q_{j-1}$ we have for each $x \in \Omega_{Y_j}$, $\mathbf{y}x \in Q_j$ (see Case (I) of the inductive definition of Q_j above). Thus, by the inductive hypothesis, $\Theta^{\pi}_j(\mathbf{y}x) = \{a_{\mathbf{y}x}\}$. The inductive definition of $a_{\mathbf{y}}$ (see Case (I) above) means that $a_{\mathbf{y}} = \sum_{x \in \Omega_{Y_j}} a_{\mathbf{y}x}$, and thus $\{a_{\mathbf{y}}\} = \sum_{x \in \Omega_{Y_j}} \{a_{\mathbf{y}x}\}$. Therefore, $\Theta^{\pi}_{j-1}(\mathbf{y}) = \sum_{x \in \Omega_{Y_j}} \Theta^{\pi}_j(\mathbf{y}x) = \sum_{x \in \Omega_{Y_j}} \{a_{\mathbf{y}x}\} = \{a_{\mathbf{y}}\}$.

Case (II). Y_j is a decision variable D_i. By definition of Θ^{π}_{j-1}, we have $\Theta^{\pi}_{j-1}(\mathbf{y}) = \Theta^{\pi}_j(\mathbf{y}d')$, where $d' \in \Omega_{D_i}$ is defined as follows: $d' = \pi_i(\mathbf{y}')$, and $\mathbf{y}'$ is tuple $\mathbf{y}$ restricted to $\mathbf{S}_i$ ($= V_j \cap \mathbf{X}$). Now, $\mathbf{y} \in \Omega_{Y_{j-1}}$ extends $\mathbf{y}'$, so by our definition of π_i, we have $\pi_i(\mathbf{y}') = d_{\mathbf{y}}$, and thus $d' = d_{\mathbf{y}}$. By the inductive hypothesis, $\Theta^{\pi}_j(\mathbf{y}d_{\mathbf{y}}) = \{a_{\mathbf{y}d_{\mathbf{y}}}\}$. By definition, $a_{\mathbf{y}d_{\mathbf{y}}} = a_{\mathbf{y}}$, so $\Theta^{\pi}_j(\mathbf{y}d_{\mathbf{y}}) = \{a_{\mathbf{y}}\}$. This shows that $\Theta^{\pi}_{j-1}(\mathbf{y}) = \Theta^{\pi}_j(\mathbf{y}d') = \{a_{\mathbf{y}}\}$.

We have proved by induction that for all $j = 0, \ldots, n$ and for all $\mathbf{y} \in Q_{j-1}$, $\Theta^{\pi}_{j-1}(\mathbf{y}) = \{a_{\mathbf{y}}\}$. In particular, we have that $\Theta^{\pi}_0(\Diamond) = \{a_\Diamond\}$, i.e., $\Theta^{\pi}_0 = \{a\}$, completing the proof of (3). $\square$

Part D: Proofs of Results in Section 9

To prove Lemma 15 we use two additional lemmas.

Lemma 39. *Let A be any subset of $\mathcal{O}$ with $\max_{\succeq}(A) = A$. Then there is at most one element $a \in A$ with $\sigma(a) \neq \pm$, i.e., with $\sigma(a) = +$ or $-$. Furthermore, if $a \in A$ is such that $\sigma(a) \neq \pm$ then for all other elements b of A, $\hat{b} < \hat{a}$. If a and b are different elements of A then $\hat{a} \neq \hat{b}$.*

Proof: The first part follows immediately from the fact that the set of elements a with $\sigma(a) \neq \pm$ forms a totally ordered subset of $\mathcal{O}$.

Suppose, to prove a contradiction, that $a \in A$ is such that $\sigma(a) \neq \pm$ and there exists some different element $b \in A$ with $\hat{b} \geq \hat{a}$. If $\sigma(a) = +$ then $a \succ b$, and if $\sigma(a) = -$ then $a \prec b$, both of which contradict $\max_{\succeq}(A) = A$. The last part follows since for any m, $\langle +, m\rangle \succ \langle \pm, m\rangle \succ \langle -, m\rangle$. $\square$

We also make use of the following result.

Lemma 40. *Let $\langle \pm, m\rangle$ and $\langle \sigma, n\rangle$ be elements of $\mathcal{O}$ with $m < n$. Then the set $\mathcal{C}(\{\langle \pm, m\rangle, \langle \sigma, n\rangle\})$ contains $\langle \pm, p\rangle$ for any p such that $m \leq p \leq n$.*

Proof: The result clearly holds if $p = m$; so, let us now assume that $p > m$. Then, we have $\langle \pm, p\rangle = (\langle +, p - m\rangle \times \langle \pm, m\rangle) + (\langle +, 0\rangle \times \langle \sigma, n\rangle)$, showing that $\langle \pm, p\rangle \in \mathcal{C}(\{\langle \pm, m\rangle, \langle \sigma, n\rangle\})$, by definition of the $\mathcal{C}(\cdot)$ operator, and since $\langle +, p - m\rangle$ and $\langle +, 0\rangle$ are in $\mathcal{O}_+$, and $\langle +, p - m\rangle + \langle +, 0\rangle = \langle +, 0\rangle = 1$. $\square$

Lemma 15. *Let A be any finite subset of $\mathcal{O}$ with $\max_{\succeq}(A) = A$. Then either $|A| = 1$ or there exists some $\sigma \in \{+, -, \pm\}$ such that $\mathcal{C}(A) = \mathcal{C}(\{\langle \pm, m\rangle, \langle \sigma, n\rangle\})$, where $m = \min\{\hat{a} : a \in A\}$, and $n = \max\{\hat{a} : a \in A\}$, and $m < n$.*

Proof: Let A be any finite subset of $\mathcal{O}$ with $\max_{\succeq}(A) = A$. Suppose that $|A| > 1$. Let a be an element of A with minimum value of $\hat{a}$, and let b be an element of B with maximum value of $\hat{b}$. Lemma 39 implies that $\hat{b} > \hat{a}$, and $a = \langle \pm, m\rangle$, where $m = \hat{a}$. Let's also write $b = \langle \sigma, n\rangle$, so that $m < n$. We will show that $\mathcal{C}(A) = \mathcal{C}(\{a, b\})$, proving the result.

Since $a, b \in A$, clearly $\mathcal{C}(A) \supseteq \mathcal{C}(\{a, b\})$, by Lemma 7(v), so we just have to prove that $\mathcal{C}(A) \subseteq \mathcal{C}(\{a, b\})$. We will first prove that $A \subseteq \mathcal{C}(\{a, b\})$. Consider any $c \in A$. If $c = b$, then clearly $c \in \mathcal{C}(\{a, b\})$, so let us now assume that $c \neq b$.

Then, by Lemma 39, $\sigma(c) = \pm$. Also, by definition of a and b, $m \leq \hat{c} \leq n$. Lemma 40 implies that $c \in \mathcal{C}(\{a, b\})$.

Finally, $A \subseteq \mathcal{C}(\{a, b\})$ implies $\mathcal{C}(A) \subseteq \mathcal{C}(\{a, b\})$, using Lemma 7(iv) and (v), completing the proof. $\square$

Lemma 17. *Consider any finite $A \subseteq \mathcal{O}$. Then A° is the unique element a of $\max_{\succeq}(A)$ with smallest value of $\hat{a}$, and $A_\circ$ is the unique element b of $\max_{\succeq}(A)$ with largest value of $\hat{b}$.*

Proof: The fact that $>^\circ$ and $>_\circ$ extend $\succ$ implies that $A_\circ$ and $A_\circ$ are in $\max_{\succeq}(A)$. First, note that if there's a unique maximal element, then the result holds trivially. So, now assume that there's more than one maximal element. Lemma 39 implies that the element a of $\max_{\succeq}(A)$ with smallest value of $\hat{a}$ has sign $\pm$. The definition of $>^\circ$ implies that $a >^\circ c$ for all $c \in \max_{\succeq}(A)$ with $\hat{c} > \hat{a}$, and thus $a = A^\circ$. Now, consider the element b of $\max_{\succeq}(A)$ with largest value of $\hat{b}$. Lemma 39 implies that for all $c \in \max_{\succeq}(A)$ apart from b have sign $\pm$, and we also have that $\hat{c} < \hat{b}$. Then the definition of $>_\circ$ implies that $b >_\circ c$, and thus $b = A_\circ$. $\square$

Proposition 13. *For finite $A \subseteq \mathcal{O}$, $A \equiv \rho(A)$.*

Proof: By Proposition 12, either (i) $A \equiv \{a\}$ for some $a \in \mathcal{O}$, and a is the unique element of $\max_{\succeq}(A)$; or (ii) there exists some $\sigma \in \{+, -, \pm\}$ and integers m, n such that $A \equiv \{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$, and $\{\langle \pm, m \rangle, \langle \sigma, n \rangle\} \subseteq \max_{\succeq}(A)$, where m is equal to $\min\{\hat{a} : a \in \max_{\succeq}(A)\}$, and $n = \max\{\hat{a} : a \in \max_{\succeq}(A)\}$, and $m < n$. First consider case (i). Lemma 17 implies that $A^\circ = A_\circ = a$, and so $\rho(A) = \{a\}$. Thus, $A \equiv \rho(A)$.

Now, consider case (ii) of Proposition 12. Along with Lemma 17, this implies that $A^\circ = \langle \pm, m \rangle$, and $A_\circ = \langle \sigma, n \rangle$, and so $A \equiv \{A^\circ, A_\circ\} = \rho(A)$. $\square$

Lemma 18. *Let A be any finite subset of $\mathcal{O}$. Then,*

(i) $(\rho(A))^\circ = A^\circ$ *and* $(\rho(A))_\circ = A_\circ$*; and*

(ii) $\rho(\rho(A)) = \rho(A)$.

Proof: (i) By definition, $A^\circ >^\circ A_\circ$ and $A_\circ >_\circ A^\circ$. This implies $(\rho(A))^\circ = (\{A^\circ, A_\circ\})^\circ$, which equals A°, and $(\rho(A))_\circ = (\{A^\circ, A_\circ\})_\circ = A_\circ$.

(ii) $\rho(\rho(A)) = \{(\rho(A))^\circ, (\rho(A))_\circ\}$, which equals $\rho(A)$ by part (i). $\square$

Lemma 19. *For any $a_1, a_2 \in \mathcal{O}$, and $q \in \mathcal{O}_+$,* *(i) if $a_1 \geq^\circ a_2$ then $q \times a_1 \geq^\circ q \times a_2$;*

(ii) if $a_1 \geq_\circ a_2$ then $q \times a_1 \geq_\circ q \times a_2$.

Proof: Write a_1, a_2 and q as $\langle \sigma, m \rangle$, $\langle \tau, n \rangle$, and $\langle +, h \rangle$,respectively,

(i) Assume that $a_1 \geq^\circ a_2$. Then either (a) $a_1 \succeq a_2$ or (b) $\sigma = \pm$ and $m < n$. If (a) then the monotonicity properties of $\succeq$ imply that $q \times a_1 \succeq q \times a_2$, and hence that $q \times a_1 \geq^\circ q \times a_2$. Otherwise, (b) $\sigma = \pm$ and $m < n$. Now, $q \times a_1 = \langle \sigma, m+h \rangle$ and $q \times a_2 = \langle \tau, n+h \rangle$, and so $q \times a_1 \geq^\circ q \times a_2$, since $\sigma = \pm$ and $m+h < n+h$.

(ii) Assume that $a_1 \geq_\circ a_2$. Then either (a) $a_1 \succeq a_2$ or (b) $\tau = \pm$ and $m > n$. If (a) then the monotonicity properties of $\succeq$ imply that $q \times a_1 \succeq q \times a_2$, and hence that $q \times a_1 \geq_\circ q \times a_2$. Otherwise, (b) $\tau = \pm$ and $m > n$. Now, $q \times a_1 = \langle \sigma, m+h \rangle$ and $q \times a_2 = \langle \tau, n+h \rangle$, and so $q \times a_1 \geq_\circ q \times a_2$, since $\tau = \pm$ and $m+h > n+h$.
$\square$

We use the following lemma to prove Lemma 20.

Lemma 41. *Let A and B be any finite subsets of $\mathcal{O}$.*

(i) $(A \cup B)^\circ = A^\circ \vee^\circ B^\circ$;

(ii) $(A \cup B)_\circ = A_\circ \vee_\circ B_\circ$;

(iii) $(q \times A)^\circ = q \times A^\circ$;

(iv) $(q \times A)_\circ = q \times A_\circ$.

Proof: (i): $(A \cup B)^\circ = \max_{>^\circ}(A \cup B)$. Since $A^\circ \vee^\circ B^\circ \in A \cup B$, we have $(A \cup B)^\circ \geq^\circ A^\circ \vee^\circ B^\circ$. Either $(A \cup B)^\circ \in A$ or $(A \cup B)^\circ \in B$. Without loss of generality assume $(A \cup B)^\circ \in A$. Then $A^\circ \vee^\circ B^\circ \geq^\circ A^\circ \geq^\circ (A \cup B)^\circ$, and so $(A \cup B)^\circ = A^\circ \vee^\circ B^\circ$. Part (ii) follows similarly.

(iii): $(q \times A)^\circ = \max_{>^\circ}(q \times A)$. Now, $q \times A^\circ \in q \times A$, so $(q \times A)^\circ \geq^\circ q \times A^\circ$. Write $(q \times A)^\circ$ as $q \times a$ for some $a \in A$. Since $A^\circ \geq^\circ a$, Lemma 19(i) implies that $q \times A^\circ \geq^\circ q \times a$. This proves (v). Part (iv) follows using exactly the same form of argument, using Lemma 19(ii). $\square$

Lemma 20. *Let A and B be finite subsets of $\mathcal{O}$.*

(i) $\rho(A \cup B) = A \veebar B = \rho(A) \veebar \rho(B)$.

(ii) $\rho(q \times A) = q \times \rho(A)$.

Proof: (i) $\rho(A \cup B) = \{(A \cup B)^\circ, (A \cup B)_\circ\}$, which, using Lemma 41, is equal to $\{A^\circ \vee^\circ B^\circ, A_\circ \vee_\circ B_\circ\}$, which equals $A \veebar B$. Lemma 18 implies that $A \veebar B = \rho(A) \veebar \rho(B)$.

(ii) $\rho(q \times A) = \{(q \times A)^\circ, (q \times A)_\circ\}$, which, by Lemma 41, equals $\{q \times A^\circ, q \times A_\circ\} = q \times \rho(A)$. $\square$

We use the following result in the proof of Lemma 21.

Lemma 42. *Let* $A^\circ = \langle \sigma, m \rangle$ *and let* $A_\circ = \langle \tau, n \rangle$. *Then either* $A^\circ = A_\circ$ *or* [$\sigma = \pm$ *and* $m < n$].

Proof: Suppose that $A^\circ \neq A_\circ$. Since $A^\circ = \max_{>^\circ}(A)$, and $>^\circ$ is a total order, $\langle \sigma, m \rangle >^\circ \langle \tau, n \rangle$, and so, by definition, either $\langle \sigma, m \rangle \succ \langle \tau, n \rangle$ or $\sigma = \pm$ and $m < n$. If it were the case that $\langle \sigma, m \rangle \succ \langle \tau, n \rangle$ then we would also have $\langle \sigma, m \rangle >_\circ \langle \tau, n \rangle$, i.e., $A^\circ >_\circ A_\circ$, which contradicts the definition of $A_\circ$ as $\max_{>_\circ}(A)$. Thus $\sigma = \pm$ and $m < n$, as required. $\square$

Lemma 21. $\rho(A) \boxplus \rho(B) = A \boxplus B \equiv \rho(A) + \rho(B)$.

Proof: By Lemma 18, $(\rho(A))^\circ = A^\circ$ and $(\rho(A))_\circ = A_\circ$, we have $\rho(A) \boxplus \rho(B) = \{A^\circ + B^\circ, A_\circ + B_\circ\} = A \boxplus B$.

First consider case when $A^\circ = A_\circ$, so that $\rho(A)$ is a singleton set $\{A^\circ\}$. Then $A \boxplus B = \{A^\circ + B^\circ, A_\circ + B_\circ\} = \{A^\circ + B^\circ, A^\circ + B_\circ\} = \rho(A) + \rho(B)$. The case when $B^\circ = B_\circ$ follows similarly.

Now consider the case when $A^\circ \neq A_\circ$ and $B^\circ \neq B_\circ$. By Lemma 42, we we can write $\rho(A)$ in the form $\{\langle \pm, m \rangle, \langle \sigma, n \rangle\}$, where $m < n$, and we can write $\rho(B)$ in the form $\{\langle \pm, g \rangle, \langle \tau, h \rangle\}$, where $g < h$. Without loss of generality, we can assume that $m \leq g$. Now, $A \boxplus B = \{\langle \pm, m \rangle, \langle \sigma, n \rangle + \langle \tau, h \rangle\}$. Also, $\rho(A) + \rho(B) = \{\langle \pm, m \rangle, \langle \sigma, n \rangle + \langle \tau, h \rangle, \langle \sigma, n \rangle + \langle \pm, g \rangle\}$. If $n < g$ then also $n < h$ and $\langle \sigma, n \rangle + \langle \tau, h \rangle = \langle \sigma, n \rangle + \langle \pm, g \rangle = \langle \sigma, n \rangle$, in which case $A \boxplus B = \rho(A) + \rho(B)$. If, on the other hand, $n \geq g$ then $\langle \sigma, n \rangle + \langle \pm, g \rangle = \langle \pm, g \rangle$, which is in $\mathcal{C}(A \boxplus B)$ (since $\langle \pm, g \rangle = (\langle +, g - m \rangle \times \langle \pm, m \rangle) + (\langle +, 0 \rangle \times (\langle \sigma, n \rangle + \langle \tau, h \rangle)))$. Thus, $A \boxplus B \subseteq \rho(A) + \rho(B) \subseteq \mathcal{C}(A \boxplus B)$. This implies, using Lemma 7(v) and (iv), that $\mathcal{C}(A \boxplus B) = \mathcal{C}(\rho(A) + \rho(B))$, which implies that $A \boxplus B \equiv \rho(A) + \rho(B)$.