Redoing the Foundations of Decision Theory

Lawrence E. Blume
Economics Department
Cornell University
leb19@cornell.edu

David A. Easley
Economics Department
Cornell University
dae3@cornell.edu

Joseph Y. Halpern*
Department of Computer Science
Cornell University
halpern@cs.cornell.edu

Abstract

In almost all current approaches to decision making, it is assumed that a decision problem is described by a set of states and set of outcomes, and the decision maker (DM) has preferences over a rather rich set of *acts*, which are functions from states to outcomes. However, most interesting decision problems do not come with a state space and an outcome space. Indeed, in complex problems it is often far from clear what the state and outcome spaces would be. We present an alternate foundation for decision making, in which the primitive objects of choice are syntactic *programs*. A program can be given semantics as a function from states to outcomes, but does not necessarily have to be described this way. A representation theorem is proved in the spirit of standard representation theorems, showing that if the DM's preference relation on programs satisfies appropriate axioms, then there exist a set S of states, a set O of outcomes, a way of viewing program as functions from S to O, a probability on S, and a utility function on O, such that the DM prefers program a to program b if and only if the expected utility of a is higher than that of b. Thus, the state space and outcome space are subjective, just like the probability and utility; they are not part of the description of the problem. A number of benefits of this approach are discussed.

1 Introduction

In almost all current approaches to decision making under uncertainty, it is assumed that a decision problem is described by a set of states and set of outcomes, and the decision maker (DM) has a preference relation on a rather rich set of *acts*, which are functions from states to outcomes. The standard representation theorems of decision theory give conditions under which the preference relation can be represented by a utility function on outcomes and numerical representation of beliefs on states. For example, Savage 1954 shows that if a DM's preference order satisfies certain axioms, then the DM's preference relation can be represented by a probability $\Pr$ on the state space and a utility function mapping outcomes to the reals such that she prefers act a to act b iff the expected utility of a (with respect to $\Pr$) is greater than that of b. Moreover, the probability measure is unique and the utility function is unique up to affine transformations. Similar representations of preference can be given with respect to other representations of uncertainty (see, for example, (Gilboa & Schmeidler 1989; Schmeidler 1989)).

Most interesting decision problems do not come with a state space and acts specified as functions on these states. Instead they are typically problems of the sort "Should I buy 100 shares of IBM or leave the money in the bank?" or "Should I attack Iraq or continue to negotiate?". To apply standard decision theory, the DM must first formulate the problem in terms of states and outcomes. But in complex decision problems, the state space and outcome space are often difficult to formulate. For example, what is the state space and outcome space in trying to decide whether to attack Iraq? And even if a DM could formulate the problem in terms of states and outcome, there is no reason to believe that someone else trying to model the problem would do it in the same way. For example, reasonable people might disagree about what facts of the world are relevant to the pricing of IBM stock. As is well known (Kahneman, Slovic, & Tversky 1982), preferences can be quite sensitive to the exact formulation. To make matters worse, a modeler may have access to information not available to the DM, and therefore incorrectly construe the decision problem from the DM's point of view.

*Supported in part by NSF under grants CTC-0208535, ITR-0325453, and IIS-0534064, by ONR under grants N00014-00-1-03-41 and N00014-01-10-511, and by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the ONR under grant N00014-01-1-0795.

Case-based decision theory (CBDT) (Gilboa & Schmeidler 2001) deals with this problem by dispensing with the state space altogether and considering instead *cases*, which are triples of the form (p, a, r), where p is a problem, a is an action and r is a result. Actions in the CBDT framework are the analogue of acts, but rather than being functions from states to outcomes, actions are primitive objects in CBDT, and are typically described in English.

We take a different view of acts here. The inspiration for our approach is the observation that objects of choice in an uncertain world have some structure to them. Individuals choose among some simple actions: "do x" or "do y". But they also perform various tests on the world and make choices contingent upon the outcome of these tests: "If the stock broker says t, do x; otherwise do y." We formalize this view by taking the objects of choice to be (syntactic) programs in a programming language. We then show that if the DM's preference relation on programs satisfies appropriate axioms, we, the modelers, can impute a state space, an outcome space, and an interpretation of programs as functions from states to outcomes such that the (induced) preferences on these functions have a subjective expected utility (SEU) representation, similar in spirit to that of Savage. Just as probability and utility are derived notions in the standard approach, and are tools that we can use to analyze and predict decisions, so too are the state space and outcome space in our framework.

This formulation of decision problems has several advantages over more traditional formulations. First, we can theorize (if we want) about only the actual observable choices available to the DM, without having states and outcomes, and without needing to view acts as functions from states to outcomes. Indeed, in the full paper, we show that we can test whether a DM's behavior is consistent with SEU despite not having states and outcomes. The second advantage is more subtle but potentially quite profound. Representation theorems are just that; they merely provide an alternative description of a preference order in terms of numerical scales. Decision theorists make no pretense that these representations have anything to do with the cognitive processes by which individuals make choices. But to the extent that the programming language models the language of the DM, we have the ability to interpret the effects of cognitive limitations having to do with the language in terms of the representation. For instance, there may be limitations on the space of acts because some sequence of tests is too computationally costly to verify. We can also take into account a DM's inability to recognize that two programs represent the same function. Finally, the approach lets us take into account the fact that different DMs use different languages to describe the same phenomena.

To understand where the state space and outcome space are coming from, first note that in our framework we need to ask two basic questions: (1) what is the programming language and (2) what is the semantics of a program; that is, what does a program *mean*. In this paper, to introduce the basic framework, we focus on a rather simple programming language, where the only construct is **if** ... **then** ... **else**. That is, if a and b are programs and t is a test, then **if** t **then** a **else** b is also a program. For example, we could have a program such as **if** the moon is in the seventh house **then** buy 100 shares of IBM **else** sell 100 shares of Microsoft. Notice that once we consider **if** ... **then** ... **else** programs, we also need to define a language of *tests*. For some of our results we also allow a randomization construct: if a_1 and a_2 are programs, then so is $ra_1+(1-r)a_2$. Intuitively, according to this program, the DM tosses a coin which lands heads with probability r. If it lands heads, the DM performs a_1; if it lands tails, the DM performs a_2. While people do not typically seem to use randomization, adding it allows us to connect our results to work in the more standard setting with states and outcomes that uses randomization, such as that of Anscombe and Aumann 1963.

There are many different ways to give semantics to programs. Here we consider what is called *input-output semantics*: a program is interpreted as a *Savage act*, i.e., a function from states to outcomes. That is, given a state space S and a outcome space O, there is a function ρ_{SO} that associates with a program a a function $\rho_{SO}(a)$ from S to O. We prove Savage-like representation theorems in this setting. We show that if the DM's preference order on programs satisfies certain axioms, then there is a state space S, an outcome space O, a probability distribution on S, a utility function on O, and a function ρ_{SO} mapping programs a to functions $\rho(a)$ from S to O such that the DM prefers act a to act b iff the expected utility of $\rho_{SO}(a)$ is at least as high as that of $\rho_{SO}(b)$. Again, we stress that the state space S and the outcome space O are subjective, just like the probability and utility; ρ_{SO} depends on S and O. If in our representation we choose a different state space S' and/or outcome space O', there is a correspondingly different function $\rho_{S'O'}$.

Besides proving that a DM's preferences can be represented by a probability and utility, Savage proves that the probability is unique and the utility is unique up to affine transformations. We cannot hope to prove such a strong uniqueness result, since for us the state space and outcome space are subjective and certainly not unique. We can show that, for the language with randomization, if acts are totally ordered (an assumption that Savage makes but we do *not* make in general), the expected utility of acts is unique up to affine transformations. However, without randomization, as we show by example, we cannot hope to get a uniqueness result. The set of acts we consider is finite, and is simply not rich enough to determine the expected utility, even up to affine transformations. To get uniqueness in the spirit of Savage, we seem to require not only randomization but a fixed outcome space.

The rest of this paper is organized as follows. In Section 2, we give the syntax and semantics of the simple programming languages we consider here, and define the notion of program equivalence. In Section 3, we state our representation theorems and discuss the (very few) postulates we needed to prove them. The key postulate turns out to be an analogue of the *cancellation* axiom (Krantz *et al.* 1971). We conclude in Section 4. Most proofs are left to the full paper.

2 The Programming Language

Before we describe programs, it is helpful to define the language of *tests*. For simplicity, we consider a simple propositional logic of tests. We start with some set T_0 of *primitive tests*. We can think of these primitive propositions as representing statements such as "the price/earnings ratio of IBM is 5", "the moon is in the seventh house" or "the economy will be strong next year". We then close off under conjunction and negation. Thus, if t_1 and t_2 are tests, then so are $t_1 \wedge t_2$ and $\neg t_1$. Let T be the set of tests that can be formed from the primitive tests in T_0 in this way. The language of tests is thus just basic propositional logic. Tests can be viewed as statements that the DM considers relevant to her decision problem. They are a part of her specification of the problem, just as states are part of the specification in the Savage and Anscombe-Aumann frameworks.

We can now describe two programming languages. In both cases we start with a set A_0 of primitive programs. We can think of the programs in $\mathcal{A}_0$ as representing such primitive actions as "buy 100 shares of IBM" or "attack Iraq". (Of course, it is up to the DM to decide what counts as primitive, both for tests and programs. It may well be that "attack Iraq" is rather complex, formed from much simpler actions.) For the first language, we simply close off the primitive programs in $\mathcal{A}_0$ under **if** ... **then** ... **else**. Thus, if a_1 and a_2 are programs and t is a test, then **if** t **then** a_1 **else** a_2 is a program. Let $\mathcal{A}_{\mathcal{A}_0,T_0}$ consist of all programs that can be formed in this way. (We omit the subscripts $\mathcal{A}_0$ and T_0 if they are either clear from context or not relevant to the discussion.) Note that $\mathcal{A}$ allows nesting, so that we can have a program such as **if** t_1 **then** a_1 **else** (**if** t_2 **then** a_2 **else** a_3).

For the second language, we close off the primitive programs in $\mathcal{A}_0$ under both **if** ... **then** ... **else** and randomization, so that if a_1 and a_2 are programs and $r \in [0, 1]$, then $ra_1 + (1 - r)a_2$ is a program. We allow arbitrary nesting of randomization and **if** ... **then** ... **else**. Let $\mathcal{A}^+_{\mathcal{A}_0,T_0}$ consist of all programs that can be formed in this way. Again, we omit subscripts when doing so does not result in loss of clarity.

We next give semantics to these languages. That is, given a state space S and an outcome space O, we associate with each program a function from S to O. The first step is to give semantics to the tests. Let π^0_S be a *test interpretation*, that is, a function associating with each primitive test a subset of S. Intuitively, $\pi^0_S(t)$ is the set of states where t is true. We then extend π^0_S in the obvious way to a function $\pi_S : T \to 2^S$ by induction on structure:

- $\pi_S(t_1 \wedge t_2) = \pi_S(t_1) \cap \pi_S(t_2)$
- $\pi_S(\neg t) = S - \pi_S(t)$.

A *program interpretation* assigns to each program a a (Savage) act, that is, a function $f_a : S \to O$. Let $\rho^0_{SO} : A_0 \to O^S$ be a program interpretation for primitive programs, which assigns to each $a_o \in A$ a function from $S \to O$. We extend ρ^0_{SO} to a function mapping all programs in A to functions from S to O by induction on structure, by defining

$$\rho_{SO}(\textbf{if } t \textbf{ then } a_1 \textbf{ else } a_2)(s) = \begin{cases} \rho_{SO}(a_1)(s) & \text{if } s \in \pi_S(t) \\ \rho_{SO}(a_2)(s) & \text{if } s \notin \pi_S(t). \end{cases} \quad (1)$$

To give semantics to the language $\mathcal{A}^+$, given S, O, and π_S, we want to associate with each act a a function from S to probability measures on O. Let $\Delta(O)$ denote the set of probability measures on O. Now, given a function $\rho^0_{SO} : \mathcal{A}_0 \to \Delta(O)^S$, we can extend it by induction on structure to all of $\mathcal{A}^+$ in the obvious way.[1] For **if** ... **then** ... **else** programs we use (1); to deal with randomization, define

$$\rho_{SO}(ra_1+(1-r)a_2))(s) = r\rho_{SO}(a_1)(s)+(1-r)\rho_{SO}(a_2)(s).$$

That is, the distribution $\rho_{SO}(ra_1 + (1 - r)a_2))(s)$ is the obvious convex combination of the distributions $\rho_{SO}(a_1)(s)$ and $\rho_{SO}(a_2)(s)$.

Note that tests and programs are amenable to two different interpretations. First, programs may not

[1] We could, as before, take ρ^0_{SO} to be a function from $\mathcal{A}$ to O^S, rather than from $\mathcal{A}$ to $\Delta(O)^S$. This would not affect our results. However, if we view general programs as functions from S to distributions over O, it seems reasonable to view primitive programs in this way as well.

necessarily be implementable (just as acts may not be implementable in the standard Savage setting). The DM may not be able to run a primitive program and may not be able to tell whether a test in a program is actually true in a given setting. However, as long as the DM in some sense "understands" the tests and programs, she might be able to determine her preferences among them. A second interpretation is that the DM has preferences only on implementable programs, that is, ones where all primitive programs can be run and the DM can determine the truth of all tests involved. Our results hold with either interpretation.

3 The Representation Theorems

We assume that the decision maker actually has two preference orders, $\succeq$ and $\succ$, on choices. Intuitively, $a \succeq b$ means that a is at least as good as b from the DM's point of view; $a \succ b$ means that a is strictly better than b. If $\succeq$ is a total order, then $\succ$ is definable in terms of $\succeq$: $a \succ b$ iff $a \succeq b$ and $b \not\succeq a$. However, in the case of partial orders (which we allow here), $\succ$ is not necessarily definable in terms of $\succeq$. Indeed while it could be the case that $a \succeq b$ and $b \not\succeq a$ implies $a \succ b$, it could also happen that the DM could say, "a is surely at least as good as b to me, but I cannot distinguish whether it is better than b, or merely as good." In this case, $a \succeq b$, $b \not\succeq a$, and $a \not\succ b$. As usual, we write $a \sim b$ if $a \succeq b$ and $b \succeq a$.

We prove various representation theorems, depending on whether there is randomization in the language, and whether the set of outcomes is taken to be fixed. One assumption we will need in every language is the irreflexivity of $\succ$.

A1. $a \not\succ a$ for all a.

For some of our results, we also assume that the preference relation is complete. In our setting, this is expressed as follows.

A2. For all a and b, either $a \succ b$ or $b \succeq a$.

A2 says that all alternatives are ranked, and if the DM is not indifferent between two alternatives, then one must be strictly preferred to the other.

The engine of our analysis is the *cancellation* postulate. Although simple versions of it have appeared in the literature (e.g. (Krantz *et al.* 1971)) it is nonetheless not well known, and so before turning to our framework we briefly explore some of its implications in more familiar settings.

3.1 The Cancellation Postulate for Choices, Savage Acts, and AA Acts

In describing the cancellation postulate, we use the notion of a *multiset*. A multiset is just a set with repetitions allowed. Thus, $\{\{a\}\}$, $\{\{a,a\}\}$, and $\{\{a,a,a\}\}$ are distinct multisets. (Note that we use the $\{\{\cdot\}\}$ to denote multisets.) Two multisets are equal if they have the same elements with the same number of occurrences. Thus $\{\{a,a,b,b,b\}\} = \{\{b,a,b,a,b\}\}$, but $\{\{a,b,a\}\} \neq \{\{a,b,\}\}$. Let C be a set of choices (not necessarily programs) among which the DM has to choose, and let $(\succeq,\succ)$ be a pair of preference relations on C.

Cancellation on C: If $\langle a_1,\dots,a_n\rangle$ and $\langle b_1,\dots,b_n\rangle$ are sequences of elements of C such that $\{\{a_1,\dots a_n\}\} = \{\{b_1,\dots,b_n\}\}$, then

1. If $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \leq n-1$, then $b_n \succeq a_n$; and
2. if, in addition, $a_k \succ b_k$ for some $k \leq n-1$, then $b_n \succ a_n$.

Roughly speaking, cancellation says that if two collections of choices are identical, then it is impossible to prefer each choice in the first collection to the corresponding choice in the second collection. In this setting, cancellation is essentially equivalent to reflexivity and transitivity.

Proposition 3.1. *A pair $(\succeq,\succ)$ of preference relations on a choice set C satisfies cancellation iff*

(a) $\succeq$ *is reflexive,*

(b) $\succ\ \subset\ \succeq$ *, and*

(c) $(\succeq,\succ)$ *is transitive; that is, if $a \succeq b$ and $b \succeq c$, then $a \succeq c$, and if either $a \succ b$ or $b \succ c$, then $a \succ c$.*

Proof: First suppose that cancellation holds. To see that $\succeq$ is reflexive, take $A = B = \{\{a\}\}$. The hypothesis of the cancellatiom axiom holds, so we must have $a \succeq a$. To see that $\succ$ is a subset of $\succeq$, suppose that $a \succ b$. Take $a_1 = b_2 = a$, $a_2 = b_1 = b$,$A = \{\{a_1,a_2\}\}$, and $B = \{\{b_1,b_2\}\}$. Since $a \succ b$, by cancellation, we must have $b \succeq a$. Thus, $\succ\ \subseteq\ \succeq$. Finally, to see that cancellation implies transitivity, consider the pair of multisets $\{\{a,b,c\}\}$ and $\{\{b,c,a\}\}$. If $a \succeq b$ and $b \succeq c$, then cancellation implies $a \succeq c$, and if one of the first two relations is strict, then $a \succ c$. We defer the proof of the converse to the full paper. ∎

In the Savage framework we strengthen the cancellation postulate:

Cancellation for Savage acts: If $\langle a_1,\dots,a_n\rangle$ and $\langle b_1,\dots,b_n\rangle$ are two sequences of Savage acts defined on a state space S such that for each state $s \in S$, $\{\{a_1(s),\dots,a_n(s)\}\} = \{\{b_1(s),\dots,b_n(s)\}\}$, then

1. if $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \leq n-1$, then $b_n \succeq a_n$; and

2. if, in addition, $a_i \succ b_i$ for some $i \le n-1$, then $b_n \succ a_n$.

Cancellation for Savage acts is a powerful assumption because equality of the multisets is required only "pointwise". Were we to require the equality of multisets for sequences of functions, then the characterization of Proposition 3.1 would apply. But we require only that for each state s, the multisets of outcomes generated by the two sequence of acts are equal; there need be no function in the second sequence equal to any function in the first sequence. Nevertheless, the conclusion still seems reasonable, because the two collections of acts deliver the same bundle of outcomes in each state.

In addition to the conditions in Proposition 3.1, Savage cancellation directly implies *event independence*, a condition at the heart of all representation theorems (and can be used to derive the Sure Thing Principle). If $T \subseteq S$, let $a_T b$ be the Savage act that agrees with a on T and with b on $S-T$; that is $a_T b(s) = a(s)$ if $s \in T$ and $a_T b(s) = b(s)$ if $s \notin T$. We say that $(\succeq, \succ)$ *satisfies event independence* if for all acts a, b, c, and c' and subsets T of the state space S, if $a_T c \succeq b_T c$, then $a_T c' \succeq b_T c'$, and similarly with $\succeq$ replaced by $\succ$.

Proposition 3.2. *If $(\succeq, \succ)$ satisfies the cancellation postulate for Savage acts then $(\succeq, \succ)$ satisfies event independence.*

Proof: Take $\langle a_1, a_2\rangle = \langle a_T c, b_T c'\rangle$ and take $\langle b_1, b_2\rangle = \langle b_T c, a_T c'\rangle$. Note that for each state $s \in T$, $\{\{a_T c(s), b_T c'(s)\}\} = \{\{a(s), b(s)\}\} = \{\{b_T c(s), a_T c'(s)\}\}$, and for each state $s \notin T$, $\{\{a_T c(s), b_T c'(s)\}\} = \{\{c(s), c'(s)\}\} = \{\{b_T c(s), a_T c'(s)\}\}$. Thus, again we can apply cancellation. ∎

Proposition 3.1 provides an axiomatic characterization of cancellation for choices. Is there a similar characterization for cancellation for Savage acts? For example, is cancellation equivalent to the combination of irreflexivity of $\succ$, the fact that $\succ$ is a subset of $\succeq$, the transitivity of $\succ$ and $\succeq$, and event independence? As the following example shows, it is not.

Example 3.3. Suppose that $S = \{s_1, s_2\}$, $O = \{o_1, o_2, o_3\}$. Let (o, o') be an abbreviation for the Savage act a such that $a(s_1) = o$ and $a(s_2) = o'$. Clearly there are nine possible acts. Suppose that $\succ$ is the transitive closure of the following string of preferences:

$$(o_1,o_1) \succ (o_1,o_2) \succ (o_2,o_1) \succ (o_2,o_2) \succ (o_3,o_1) \\ \succ (o_1,o_3) \succ (o_2,o_3) \succ (o_3,o_2) \succ (o_3,o_3);$$

let $\succeq$ be the reflexive closure of $\succ$. By construction, $\succ$ is irreflexive, $\succ$ is a subset of $\succeq$, $\succ$ and $\succeq$ are transitive. To see that $(\succeq, \succ)$ satisfies event independence, note that

- $(x, o_1) \succ (x, o_2) \succ (x, o_3)$ for $x \in \{o_1, o_2, o_3\}$;
- $(o_1, y) \succ (o_2, y) \succ (o_3, y)$ for $y \in \{o_1, o_2, o_3\}$.

However, cancellation for Savage acts does not hold, since $(o_1, o_2) \succ (o_2, o_1)$, $(o_2, o_3) \succ (o_3, o_2)$, and $(o_3, o_1) \succ (o_1, o_3)$. ∎

We do not know if cancellation for Savage acts has a simple characterization. However, once we allow randomization, we can get an elegant characterization of cancellation. As usual, define an *AA act* (for Anscombe-Aumann) to be a function from states to lotteries over outcomes. We can define an analogue of cancellation for AA acts. Note that since an AA act is a function from states to lotteries over outcomes, if a and b are acts, $a+b$ is a well-defined function on states: $(a+b)(s)(o) = a(s)(o) + b(s)(o)$. While $(a+b)(s)$ is not a lottery on outcomes (since its range is $[0,2]$, not $[0,1]$), it can be viewed as an "unnormalized lottery".

Cancellation for AA acts: If $\langle a_1, \ldots, a_n\rangle$ and $\langle b_1, \ldots, b_n\rangle$ are two sequences of AA acts such that $\sum_{i=1}^n a_i = \sum_{i=1}^n b_i$, then

1. if $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \le n-1$, then $b_n \succeq a_n$; and
2. if, in addition, $a_i \succ b_i$ for some $i \le n-1$, then $b_n \succ a_n$.

Note that cancellation for AA acts can be viewed as a generalization of cancellation for Savage acts. In the case of Savage acts, the probabilities are just point masses, so the fact that $(\sum_{i=1}^n a_i)(s) = (\sum_{i=1}^n b_i)(s)$ for all states s says that the multisets of outcomes must be equal for all states s.

For AA acts, there is a standard probabilistic independence axiom. We say that the preference orders $(\succeq, \succ)$ satisfy *AA act independence* if for all AA acts a, b, and c, and all $r \in (0,1]$, we have $a \succeq b$ iff $ra + (1-r)c \succeq rb + (1-r)c$ and $a \succ b$ iff $ra + (1-r)c \succ rb + (1-r)c$. We say that $(\succeq, \succ)$ satisfies *rational AA act independence* if they satisfy AA act independence for all rational $r \in (0,1]$.

Proposition 3.4. *If $(\succeq, \succ)$ satisfies* **A2** *and cancellation for AA acts, then $(\succeq, \succ)$ satisfies rational AA act independence.*

Proof: Suppose that $a \succeq b$ and $r = m/n$. Let $a_1 = \cdots = a_m = a$ and $a_{m+1} = \cdots = a_{m+n} = rb + (1-r)c$; let $b_1 = \cdots = b_m = b$ and $b_{m+1} = \cdots = b_{m+n} = ra + (1-r)c$. It is easy to check that $\sum_{i=1}^{m+n} a_i = \sum_{i=1}^{m+n} b_i$. If $rb + (1-r)c \succ ra + (1-r)c$, then we get a contradiction to cancellation for AA actions. Thus, by **A2**, we must have

$ra + (1-r)c \succeq rb + (1-r)c$. The same collection of programs can be used for the converse implication as well as for the result with $\succ$. ∎

We can now characterize cancellation for AA acts for total orders.

Theorem 3.5. *If $(\succeq, \succ)$ is a pair of preference relations satisfying* **A2***, then $(\succeq, \succ)$ satisfies cancellation for AA acts iff $\succ$ is irreflexive, $\succ \subseteq \succeq$, $(\succeq, \succ)$ is transitive, and $(\succeq, \succ)$ satisfies rational AA act independence.*

Proof: The fact that extended cancellation implies that $(\succeq, \succ)$ has the required properties follows from Propositions 3.1, and 3.4. For the converse, suppose that $\langle a_1, \ldots, a_n \rangle$ and $\langle b_1, \ldots, b_n \rangle$ are sequences of AA acts such that $a_1 + \cdots + a_n = b_1 + \cdots + b_n$ and $a_i \succeq b_i$ for $i = 1, \ldots, n-1$. By way of contradiction, suppose that $a_n \succ b_n$. Let $c = \frac{1}{n-1}(a_2 + \cdots + a_n))$. Since $a_1 \succeq b_1$, by rational AA act independence, we get that

$$\begin{array}{ll} & \frac{1}{n}(a_1 + \cdots + a_n) \\ = & \frac{1}{n}a_1 + \frac{n-1}{n}c \succeq \frac{1}{n}b_1 + \frac{n-1}{n}c \\ = & \frac{1}{n}(b_1 + a_2 + \cdots + a_n). \end{array}$$

By induction (using transitivity) and the fact that $a_n \succ b_n$, it follows that $\frac{1}{n}(a_1 + \cdots + a_n) \succ \frac{1}{n}(b_1 + \cdots + b_{n-1} + b_n)$. But this contradicts the assumption that $a_1 + \cdots + a_n = b_1 + \cdots + b_n$. Since it is not the case that $a_n \succ b_n$, by **A2**, we must have $b_n \succeq a_n$, as desired. If $a_i \succ b_i$ for some $i \in \{1, \ldots, n-k\}$, a similar argument shows that $b_n \succ a_n$. ∎

It seems that **A2** plays a critical role in the proof of Proposition 3.4 and hence Theorem 3.5. It turns out that by strengthening cancellation appropriately, we can avoid this use of **A2**. We first state the strengthened cancellation postulate for Savage acts, since we shall need it later, and then for AA acts.

> **Extended cancellation for Savage acts:** If $\langle a_1, \ldots, a_n \rangle$ and $\langle b_1, \ldots, b_n \rangle$ are two sequences of Savage acts defined on a state space S such that for each state $s \in S$ we have $\{\{a_1(s), \ldots, a_n(s)\}\} = \{\{b_1(s), \ldots, b_n(s)\}\}$, then
> 1. if there exists some $k < n$ such that $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \leq k$, $a_{k+1} = \ldots = a_n$, and $b_{k+1} = \ldots = b_n$, then $b_n \succeq a_n$; and
> 2. if, in addition, $a_i \succ b_i$ for some $i \leq k$, then $b_n \succ a_n$.

Clearly, cancellation for Savage acts is just the special case of extended cancellation where $k = n-1$. The intuition behind the extended cancellation postulate is identical to that for the basic cancellation postulate; moreover, it is easy to see that the existence of an SEU representation for $\succeq$ implies extended cancellation.

There is an obvious analogue of extended cancellation for AA acts:

> **Extended cancellation for AA acts:** If $\langle a_1, \ldots, a_n \rangle$ and $\langle b_1, \ldots, b_n \rangle$ are two sequences of AA acts such that $\sum_{i=1}^n a_i = \sum_{i=1}^n b_i$, then
> 1. if there exists some $k < n$ such that $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \leq k$, $a_{k+1} = \ldots = a_n$, and $b_{k+1} = \ldots = b_n$, then $b_n \succeq a_n$; and
> 2. if, in addition, $a_i \succ b_i$ for some $i \leq k$, then $b_n \succ a_n$.

Extended cancellation is just what we need to remove the need for **A2** in Proposition 3.4.

Proposition 3.6. *If $(\succeq, \succ)$ satisfies extended cancellation for AA acts, then $(\succeq, \succ)$ satisfies rational AA act independence.*

Proof: The proof is identical to that of Proposition 3.4, except that we can show that $ra + (1-r)c \succeq rb + (1-r)c$ immediately using extended cancellation, without invoking **A2**. We leave details to the reader. ∎

We can now get the desired characterization of cancellation for AA acts.

Theorem 3.7. *$(\succeq, \succ)$ satisfies extended cancellation for AA acts iff $\succ$ is irreflexive, $\succ \subseteq \succeq$, $(\succeq, \succ)$ is transitive, and $(\succeq, \succ)$ satisfies rational AA act independence.*

Proof: The "if" direction is immediate from Propositions 3.1, and 3.6. For the converse, we proceed much as in the proof of Theorem 3.5. We leave details to the full paper. ∎

3.2 The Cancellation Postulate for Programs

We use cancellation to get a representation theorem for preference orders on programs. However, the definition of the cancellation postulates for Savage acts and AA acts make heavy use of states. We now show how we can get an analogue of this postulate for programs.

Definition 3.8. Given a set $T_0 = \{t_1, \ldots, t_n\}$ of primitive tests, an *atom over* T_0 is a hypothesis of the form $t'_1 \wedge \ldots \wedge t'_n$, where t'_i is either t_i or $\neg t_i$.

An atom δ can be identified with the truth assignment v_δ to the primitive tests such that $v_\delta(t_i) = \mathbf{true}$

iff t_i appears unnegated in δ. If there are n primitive tests in T_0, there are 2^n atoms. Let $At(T_0)$ denote the set of atoms over T_0. It is easy to see that, for all tests $t \in T$ and atoms $\delta \in At(T_0)$, either $\pi_S(\delta) \subseteq \pi_S(t)$ for all state spaces S and interpretations π_S or $\pi_S(\delta) \cap \pi_S(t) = \emptyset$ for all state spaces S and interpretations π_S. (The formal proof is by induction on the structure of t.) We write $\delta \Rightarrow t$ if the former is the case.

A program in $\mathcal{A}$ can be identified with a function from atoms to primitive programs in an obvious way. For example, if a_1, a_2, and a_3 are primitive programs and $T_0 = \{t_1, t_2\}$, then the program $a =$ **if** t_1 **then** a_1 **else** (**if** t_2 **then** a_2 **else** a_3) can be identified with the function f_a such that

- $f_a(t_1 \wedge t_2) = f_a(t_1 \wedge \neg t_2) = a_1$;
- $f_a(\neg t_1 \wedge t_2) = a_2$; and
- $f_a(\neg t_1 \wedge \neg t_2) = a_3$.

Formally, we define f_a by induction on the structure of programs. If $a \in \mathcal{A}_0$, then f_a is the constant function a, and

$$f_{\textbf{if } t \textbf{ then } a \textbf{ else } b}(\delta) = \begin{cases} f_a(\delta) & \text{if } \delta \Rightarrow t \\ f_b(\delta) & \text{otherwise.} \end{cases}$$

The cancellation postulate that we use for the language $\mathcal{A}_0$ is simply extended cancellation for Savage acts, with atoms playing the role of states:

A3. If $\langle a_1, \ldots, a_n \rangle$ and $\langle b_1, \ldots, b_n \rangle$ are two sequences of programs in $\mathcal{A}_{\mathcal{A}_0, T_0}$, and for each atom δ over T_0, $\{\{f_{a_1}(\delta), \ldots, f_{a_n}(\delta)\}\} = \{\{f_{b_1}(\delta), \ldots, f_{b_n}(\delta)\}\}$, then

1. if there exists some $k < n$ such that $a_i \succeq b_i$ or $a_i \succ b_i$ for $i \leq k$, $a_{k+1} = \ldots = a_n$, and $b_{k+1} = \ldots = b_n$, then $b_n \succeq a_n$; and
2. if, in addition, $a_i \succ b_i$ for some $i \leq k$, then $b_n \succ a_n$.

Of course, we can prove analogues of Propositions 3.1 and 3.2 using **A3**. **A3** has another consequence when choices are programs: a DM must be indifferent between equivalent programs, where programs a and b are *equivalent* if, no matter what interpretation is used, they are interpreted as the same function. For example, **if** t **then** a **else** b is equivalent to **if** $\neg t$ **then** b **else** a; no matter what the test t and programs a and b are, these two programs have the same input-output semantics. Similarly, if t and t' are equivalent tests, then **if** t **then** a **else** b is equivalent to **if** t' **then** a **else** b.

Definition 3.9. Programs a and b are *equivalent*, denoted $a \equiv b$, if, for all S, O, π_S, and ρ_{SO}, we have $\rho_{SO}(a) = \rho_{SO}(b)$.

The general problem of checking whether two programs are equivalent is at least as hard as checking whether two propositional formulas are equivalent, and so is co-NP-hard. It is not hard to show that it is in fact co-NP-complete. Nevertheless, it is a consequence of cancellation that a DM must be indifferent between two equivalent programs.

Proposition 3.10. *Suppose that* $(\succeq, \succ)$ *satisfies* **A3**. *Then* $a \equiv b$ *implies* $a \sim b$.

Proof: It is easy to check that if $a \equiv b$ then $f_a = f_b$. Let S be $At(T_0)$, the set of atoms, let O be $\mathcal{A}_0$, the set of primitive programs, and define $\rho^0_{SO}(c)$ to be the constant function c for a primitive program c. It is easy to see that $\rho_{SO}(c) = f_c$ for all programs c. Since if $a \equiv b$, then $\rho_{SO}(a) = \rho_{SO}(b)$, we must have $f_a = f_b$. Now apply **A3** with $a_1 = a$ and $b_1 = b$ to get $b \succeq a$, and then reverse the roles of a and b. ∎

To get a representation theorem for $\mathcal{A}^+$, we use the cancellation postulate for AA acts, again replacing states by atoms. The idea now is that we can identify each program a with a function f_a mapping atoms into distributions over primitive programs. For example, if t is the only test, then the program $a = \frac{1}{2}a_1 + \frac{1}{2}(\textbf{if } t \textbf{ then } a_2 \textbf{ else } a_3)$ can be identified with the function f_a such that

- $f_a(t)(a_1) = 1/2$; $f_a(t)(a_2) = 1/2$
- $f_a(\neg t)(a_1) = 1/2$; $f_a(\neg t)(a_2) = 1/2$.

Formally, we just extend the definition of f_a given in the previous section by defining

$$f_{ra_1+(1-r)a_2}(t) = rf_{a_1}(t) + (1-r)f_{a_2}(t).$$

We then get the obvious analogue of extended cancellation for AA acts:

A3′. If $\langle a_1, \ldots, a_n \rangle$ and $\langle b_1, \ldots, b_n \rangle$ are two sequences of acts in $\mathcal{A}^+_{\mathcal{A}_0, T_0}$ such that $f_{a_1} + \ldots + f_{a_n} = f_{b_1} + \ldots + f_{b_n}$, then if $a_i \succeq b_i$ for $i = 1, \ldots, k$, where $k < n$, and $a_{k+1} = \ldots = a_n$, and $b_{k+1} = \ldots = b_n$, then $b_n \succeq a_n$. Moreover, if $a_i \succ b_i$ for some $i \in \{1, \ldots, n-k\}$, then $a_n \succ b_n$.

Again, **A3′** can be viewed as a generalization of **A3**.

Theorem 3.7 shows that we can replace **A3′** by **A1** and postulates that say that (a) $\succeq \subseteq \succ$, (b) $(\succeq, \succ)$ is transitive, (c) $(\succeq, \succ)$ satisfies rational AA act independence, and (d) $a \equiv b$ implies $a \sim b$.

3.3 A Representation Theorem for $\mathcal{A}$

We are now ready to state our first representation theorem. In the theorem, if f is a Savage act mapping states S to outcomes O and u is a utility function on

O (so that $u : O \rightarrow I\!R$), then $u_f : S \rightarrow I\!R$ is the function defined by taking $u_f(s) = u(f(s))$. If $\Pr$ is a probability measure on S, then $\mathrm{E}_{\Pr}(u_f)$ is the expectation of u_f with respect to $\Pr$.

Theorem 3.11. *If $(\succeq, \succ)$ are preference orders on acts in $\mathcal{A}_{\mathcal{A}_0,T_0}$ satisfying* **A1** *and* **A3**, *then there exist a finite set S of states, a set $\mathcal{P}$ of probability measures on S, a finite set O of outcomes, a set $\mathcal{U}$ of utility functions on O, a set $\mathcal{V} \subseteq \mathcal{P} \times \mathcal{U}$, a test interpretation π^0_S, and a program interpretation ρ^0_{SO} such that $a \succeq b$ iff $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) \le \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$ for all $(\Pr, u) \in \mathcal{V}$ and $a \succ b$ iff for $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) < \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$ for all $(\Pr, u) \in \mathcal{V}$. Either $\mathcal{P}$ or $\mathcal{U}$ can be taken to be a singleton. Moreover, if* **A2** *also holds, then $\mathcal{V}$ can be taken to be a singleton and S can be taken to be $At(T_0)$.*

Note that, in Theorem 3.11, there are no uniqueness requirements on $\mathcal{P}$ or $\mathcal{U}$. In part, this is because the state space and outcome space are not unique. But even if **A2** holds, so that the state space can be taken to be the set of atoms, the probability and the utility are far from unique, as the following example shows.

Example 3.12. Take $\mathcal{A}_0 = \{a, b\}$ and $T_0 = \{t\}$. Suppose that $\succeq$ is the reflexive transitive closure of the following string of preferences:

$$a \succ \textbf{if } t \textbf{ then } a \textbf{ else } b \succ \textbf{if } t \textbf{ then } b \textbf{ else } a \succ b.$$

It is not hard to check that every program in $\mathcal{A}$ is equivalent to one of these four, so **A2** holds, and we can take the state space to be $S^* = \{t, \neg t\}$. Let $O^* = \{o_1, o_2\}$, and define $\rho^0_{S^*O^*}$ so that $\rho^0_{S^*O^*}(a)$ is the constant function o_1 and $\rho^0_{S^*O^*}(b)$ is the constant function o_2. Now define $\pi^0_{S^*}$ in the obvious way, so that $\pi^0_{S^*}(t) = \{t\}$ and $\pi^0_{S^*}(\neg t) = \{\neg t\}$. We can represent the preference order by using any probability measure $\Pr^*$ such that $\Pr^*(s_1) > \Pr^*(s_2)$ and utility function u^* such that $u^*(o_1) > u^*(o_2)$. ∎

As Example 3.12 shows, the problem really is that the set of actions is not rich enough to determine the probability and utility. By way of contrast, Savage's postulates ensure that the state space is infinite and that there are at least two outcomes. Since the acts are all functions from states to outcomes, there must be uncountably many acts in Savage's framework.

The next example shows that if the order is partial and we want the representation to involve just a single utility function, then we cannot take the state space to be the set of atoms.

Example 3.13. Suppose that $T_0 = \emptyset$, and $\mathcal{A}_0$ (and hence $\mathcal{A}$) consists of the two primitive programs a and b, which are incomparable. In this case, the smallest state space we can use has cardinality at least 2. For if $|S| = 1$, then there is only one possible probability measure on S, so a and b cannot be incomparable. Since there is only one atom when there are no primitive propositions, we cannot take the state space to be the set of atoms. (There is nothing special about taking $T_0 = \emptyset$ here; similar examples can be constructed for arbitrary choices of T_0.) It is also immediate that there is no representation where the outcomes space has only one element. There is a representation where the state and outcome space have cardinality 2: let $S = \{s_1, s_2\}$ and $O = \{o_1, o_2\}$; define ρ^0_{SO} so that $\rho^0_{SO}(a)(s_i) = o_i$ and $\rho^0_{SO}(b)(s_i) = o_{i\oplus 1}$, for $i = 1, 2$ (where $\oplus$ represents addition mod 2); let $\mathcal{P}$ be any set of probability measures that includes measures $\Pr_1$ and $\Pr_2$ such that $\Pr_1(s_1) > \Pr_1(s_2)$ and $\Pr_2(s_2) > \Pr_2(s_1)$; let $\mathcal{U}$ be any set of utility functions that includes a utility function such that $u(o_1) \neq u(o_2)$. It is easy to see that this choice gives us a representation of the preference order that makes a and b incomparable. This suggests that there are no interesting uniqueness requirements satisfied by $\mathcal{P}$ and $\mathcal{U}$. ∎

We remark that it follows from the proof of Theorem 3.11 that, even if the order is partial, there is a representation involving a single probability measure (and possibly many utility functions) such that the state space is $At(T_0)$.

3.4 A Representation Theorem for $\mathcal{A}^+$

A3′ does not suffice to get a representation theorem for the richer language $\mathcal{A}^+$. As we have observed **A3′** gives us rational AA act independence. To get a representation theorem, we need to have act independence even for acts with real coefficients as well. Moreover, we need a standard Archimedean property. The following two postulates do the job.

A4. If $r \in (0, 1]$, then $a \succeq b$ iff $ra + (1-r)c \succeq rb + (1-r)c$ and $a \succ b$ iff $ra + (1-r)c \succ rb + (1-r)c$.

A5. If $a \succ b \succ c$ then there exist $r, r' \in [0, 1]$ such that $a \succ ra + (1-r)c \succ b \succ r'a + (1-r')c \succ c$.

We now get the following analogue of Theorem 3.11. With randomization, if **A2** holds, we get some degree of uniqueness. Although the state space, outcome space, probability, and utility are not unique, expected utilities are unique up to affine transformations. To make our discussion of uniqueness clearer, say that $(S, O, \mathcal{P}, \pi^0_S, \rho^0_{SO}, u)$ *is a representation of* $(\succeq, \succ)$ if it satisfies the conditions of Theorem 3.11. Note that if a is a program, then $\rho_{SO}(a)$ is an AA act. As usual, we take $E_{\Pr}(u_{\rho_{SO}(a)})$, the expected utility of this AA act with respect to $\Pr$, to be

$$\sum_{s \in S} \sum_{o \in O} Pr(s) u(o) (\rho_{SO}(a))(s)(o).$$

Theorem 3.14. *If $(\succeq, \succ)$ are preference orders on acts in $\mathcal{A}^+_{\mathcal{A}_0,T_0}$ satisfying* **A1**, **A3′**, **A4**, *and* **A5**, *then there exist a finite set S of states, a set $\mathcal{P}$ of probability measures on S, a finite set O of outcomes, a set $\mathcal{U}$ of utility functions on O, a set $\mathcal{V} \subseteq \mathcal{P} \times \mathcal{U}$, a test interpretation π^0_S, and a program interpretation ρ^0_{SO} such that $a \succeq b$ iff $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) \geq \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$ for all $(\Pr, u) \in \mathcal{V}$ and $a \succ b$ iff $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) > \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$ for all $(\Pr, u) \in \mathcal{V}$. Either $\mathcal{P}$ or $\mathcal{U}$ can be taken to be a singleton. Moreover, if* **A2** *also holds, then $\mathcal{V}$ can be taken to be a singleton, S can be taken to be $At(T_0)$, and if $(S, O, \Pr, \pi^0_S, \rho^0_{SO}, u)$ and $(S', O', \Pr', \pi^0_{S'}, \rho^0_{S'O'}, u')$ both represent $(\succeq, \succ)$, then there exist constants α and β such that for all acts $a \in \mathcal{A}^+_{\mathcal{A}_0,T_0}$, $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) = \alpha \mathrm{E}_{\Pr'}(u'_{\rho_{S'O'}(a)}) + \beta$.*

If **A2** holds, then Theorem 3.14 says that the expected utility is unique up to affine transformation, but makes no uniqueness claims for either the probability or the utility. This is not surprising, given that, in general, the probability and utility will be over quite different spaces. But even if two representations use the same state and outcome spaces, not much can be said, as the following example shows.

Example 3.15. Suppose that $\mathcal{A}_0 = \{a, b\}$, $T_0 = \{t\}$, and $(\succeq, \succ)$ is a pair of preference relations satisfying **A1**, **A2**, **A3′**, **A4**, and **A5** such that $a \succ b$ and **if** t **then** a **else** $b \sim \frac{1}{2}a + \frac{1}{2}b$. It is easy to see that these constraints completely determine $(\succeq, \succ)$. Let $S = At(T_0)$, $O = \{a_t, a_{\neg t}, b_t, b_{\neg t}\}$, $\pi^0_S(t) = \{t\}$, $\pi^0_S(\neg t) = \{\neg t\}$, and $\rho_{SO}(c)(s) = c_s$ for $c \in \{a, b\}$ and $s \in \{t, \neg t\}$. There are many representations of $(\succeq, \succ)$ with this state and outcome space: for example, we could take $\Pr_1(t) = 1/2$, $u_1(a_t) = u_1(a_{\neg t}) = 1$, and $u_1(b_t) = u_1(b_{\neg t}) = 0$; or we could take $\Pr_2(t) = 3/4$, $u_2(a_t) = 2/3$, $u_2(a_{\neg t}) = 2$, and $u_2(b_t) = -u_2(b_{\neg t}) = 0$. We leave it to the reader to check that these choices both lead to representations that give the same expected utility to acts. ∎

As in the case of the language $\mathcal{A}$, we cannot in general take the state space to be the set of atoms. Specifically, if $\mathcal{A}_0$ consists of two primitive programs, and we take all programs in $\mathcal{A}^+_0$ to be incomparable, then the same argument as in Example 3.13 shows that we cannot take S to be $At(T_0)$, and there are no interesting uniqueness requirements that we can place on the set of probability measures or the utility function.

3.5 Objective Outcomes

In many applications it seems reasonable to consider the outcome space to be objective, rather than subjective. For example, if we are considering decisions involving trading in securities, we can take the outcome space to be dollar amounts. Given a fixed, finite set O of outcomes, we consider the set of acts that result when, in addition to the primitive acts in $\mathcal{A}_0$, we assume that there is a special act a_o for each outcome $o \in O$. Call the resulting language $\mathcal{A}_{\mathcal{A}_0,T_0,O}$ or $\mathcal{A}^+_{\mathcal{A}_0,T_0,O}$, depending on whether we allow randomization. We then define ρ_{SO} so that $\rho_{SO}(a_o)$ is interpreted as the constant function o for each outcome o.

With an objective outcome space O, we have not been able to prove a representation theorem for $\mathcal{A}$, but we can get a representation theorem for $\mathcal{A}^+$ that is quite close to that of Savage and Anscombe and Aumann. We need a postulate that, roughly speaking, ensures that all outcomes o are treated the same way in all states. This postulate is the obvious analogue of Savage's state independence postulate. Given a test t, we write $a \succeq_t b$ if for some (and hence all, given cancellation) acts c, we have

$$\textbf{if } t \textbf{ then } a \textbf{ else } c \succeq \textbf{if } t \textbf{ then } b \textbf{ else } c.$$

We say that t is *null* if, for all a, b, and c, we have

$$\textbf{if } t \textbf{ then } a \textbf{ else } c \sim \textbf{if } t \textbf{ then } b \textbf{ else } c.$$

Define a *generalized outcome act* to be a program of the form $ra_{o_1} + (1-r)a_{o_2}$, where o_1 and o_2 are outcomes.

A6. If t is not null and a_1 and a_2 are generalized outcomes, then $a_1 \succeq a_2$ iff $a_1 \succeq_t a_2$ and similarly with $\succeq$ replaced by $\succ$.

A7. There exist outcomes o_0 and o_1 such that $a_{o_1} \succ a_{o_0}$.

Theorem 3.16. *If $(\succeq, \succ)$ are preference orders on acts in $\mathcal{A}^+_{\mathcal{A}_0,T_0,O}$ satisfying* **A1**, **A3′**, *and* **A4**-**A7**, *then there exist a set S of states, a set $\mathcal{P}$ of probability measures on S, a set $\mathcal{U}$ of utility functions on O, a set $\mathcal{V} \subseteq \mathcal{P} \times \mathcal{U}$, a test interpretation π^0_S, and a program interpretation ρ^0_{SO} such that $a \succeq b$ iff for all $(\Pr, u) \in \mathcal{V}$, $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) \geq \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$ and $a \succ b$ iff for all $(\Pr, u) \in \mathcal{V}$, $\mathrm{E}_{\Pr}(u_{\rho_{SO}(a)}) > \mathrm{E}_{\Pr}(u_{\rho_{SO}(b)})$. If* **A2** *holds, then we can take $\mathcal{V}$ to be a singleton, the probability is uniquely determined, and the utility function is determined up to affine transformations. That is, if $(S, \Pr, \pi^0_S, \rho^0_{SO}, u)$ and $(S', \Pr', \pi^0_{S'}, \rho^0_{S'O}, u')$ are two representations of the preference order, then, for all tests t, $\Pr(\pi_S(t)) = \Pr'(\pi_{S'}(t))$, and there exist α and β such that $u' = \alpha u + \beta$.*

Note that in Theorem 3.16, even if acts are totally ordered, we cannot take the state space to consist only of atoms as the following example shows.

Example 3.17. Suppose that there are two outcomes: o_1 (\$1,000) and o_0 (\$0), and one primitive act a: buying 10 shares of IBM. Act a intuitively can return somewhere between \$0 and \$1,000; thus, $o_1 \succ a \succ o_0$. There are no tests. If there were a representation with only one state, say s, in state s, a must return either \$1,000 or \$0. Whichever it is, we cannot represent the preference order. ∎

Our earlier representation theorems always involved a single utility function. As the following example shows, we can use neither a single utility function nor a single probability measure here. Moreover, there cannot be a representation where the set of probability-utility pairs has the form $\mathcal{P} \times \mathcal{U}$.

Example 3.18. Suppose that $O = \{o_1, o_0, o\}$, $\mathcal{A}_0 = \emptyset$, and $T_0 = \{t\}$. Let u_1 be a utility function such that $u_1(o_1) = 1$, $u_1(o_0) = 0$, and $u_1(o) = 3/4$; let u_2 be a utility function such that $u_2(o_1) = 1$, $u_2(o_0) = 0$, and $u_2(o) = 1/4$; let $\Pr_1$ be a probability measure on $S = At(T_0)$ such that $\Pr_1(t) = 1/4$, and let $\Pr_2$ be a probability measure on S such that $\Pr_2(t) = 3/4$. Consider the preference order on $\mathcal{A}^+_{\mathcal{A}_0,T_0,O}$ generated from $\mathcal{V} = \{(\Pr_1, u_1), (\Pr_2, u_2)\}$, taking $\pi_S(t) = \{t\}$. It is easy to see that this preference order has the following properties:

- $a_{o_1} \succ a_o \succ a_{o_0}$;
- a_o and $\frac{1}{2}a_{o_1} + \frac{1}{2}a_{o_0}$ are incomparable;
- $(\textbf{if } t \textbf{ then } a_{o_1} \textbf{ else } a_{o_0})$ and $\frac{1}{2}a_{o_1} + \frac{1}{2}a_{o_0}$ are incomparable; and
- $(\textbf{if } t \textbf{ then } a_{o_1} \textbf{ else } a_{o_0}) \sim a_o$.

Consider a representation of this order. It is easy to see that to ensure that a_o is not comparable to $\frac{1}{2}a_{o_1} + \frac{1}{2}a_{o_0}$, the representation must have utility functions u_1 and u_2 such that $u_1(o) > \frac{1}{2}(u_1(o_1) + u_1(o_0))$ and $u_2(o) > \frac{1}{2}(u_2(o_1) + u_2(o_0))$. To ensure that $\textbf{if } t \textbf{ then } a_{o_1} \textbf{ else } a_{o_0}$ and $\frac{1}{2}a_{o_1} + \frac{1}{2}a_{o_0}$ are incomparable, there must be two probability measures $\Pr_1$ and $\Pr_2$ in the representation such that $\Pr_1(\pi_S(t)) < 1/2$ and $\Pr_2(\pi_S(t)) > 1/2$. Finally, to ensure that $(\textbf{if } t \textbf{ then } a_o \textbf{ else } a_{o_0}) \sim a_o$, we cannot have both $(\Pr_1, u_1)$ and $(\Pr_1, u_2)$ in $\mathcal{V}$. ∎

4 Conclusions

Most critiques of Bayesian decision making have left two assumptions unquestioned: that beliefs may be represented with a single number, and that all possible states and outcomes are known beforehand. The work presented here directly addresses these concerns. We have shown that by viewing acts as programs rather than as functions from states to outcomes, we can prove results much in the spirit of the well-known representation theorems of Savage and Anscombe and Aumann; the main difference is that the state space and outcome space are now objective, rather than being given as part of the decision problem. So what does all this buy us? To the extent that we can prove only such representation theorems, the new framework does not buy us much (although we still claim that thinking of acts as programs is typically more natural than thinking of them as functions—rather than thinking about a state and outcome space, a DM need just think of what he/she can do). We have proved these results just to show how our approach relates to the standard approaches. We believe that the real benefit of our approach will be realized when we move beyond the limited setting we have considered in this paper. We list some possibilities here:

- Once we move to game-theoretic settings with more than one agent, we can allow different agents to use different languages. For example, when trying to decide whether to buy 100 shares of IBM, one agent can consider quantitative issues like price/earnings ratio, while another might consult astrological tables. The agent who uses astrology might not understand price/earnings ratios (the notion is simply not in his vocabulary) and, similarly, the agent who uses quantitative methods might not understand what it means for the moon to be in the seventh house. Nevertheless, they can trade, as long as they both have available primitive actions like "buy 100 shares of IBM" and "sell 100 shares of IBM". Agreeing to disagree results (Aumann 1976), which say that agents with a common prior must have a common posterior (they cannot agree to disagree) no longer hold. Not only is there no common prior, once the state space is subjective, the agents do not even have a common state space on which to put a common prior. Moreover, notions of unawareness (Fagin & Halpern 1988; Heifetz, Meier, & Schipper 2003; Halpern & Rêgo 2005; Modica & Rustichini 1999) come to the fore. Agents need to reason about the fact that other agents may be aware of tests of which they are unaware. See (Feinberg 2004; 2005; Halpern & Rêgo 2006) for preliminary work taking awareness into account in games.
- We focus here on static, one-shot decisions. Dealing with decision-making over time becomes more subtle. Learning becomes not just a matter of conditioning, but also learning about new notions (i.e., expanding the language of tests). Note that we can think of learning unanticipated events as a combination of learning about a new notion and then conditioning. This framework lends itself naturally to vocabulary expansion—it just amounts to expand-

ing the possible set of programs.

- We have considered only a very simple programming language with input-output semantics. Interesting new issues arise once we consider richer programming languages. For example, suppose that we allow concatenation of programs, so that if a and b are programs, then so is $a; b$. Intuitively, $a; b$ means "do a, and then do b". We can still use input-output semantics for this richer programming language, but it might also be of interest to consider a different semantics, where we associate with a program the sequence of states it goes through before reaching the outcome. This "path semantics" makes finer distinctions than input-output semantics; two programs that have the same input-output behavior might follow quite different paths to get to an outcome, starting at the same initial state. The framework thus lets us explore how different choices of semantics for programs affect an agent's preference order.
- As we have seen (Proposition 3.10), cancellation forces a DM to be indifferent between two equivalent programs. But since testing program equivalence is co-NP-complete, it is unreasonable to expect that agents will necessarily be indifferent between two equivalent programs. It would be very interesting to consider weakenings of the cancellation axiom that do not force a DM to be indifferent between equivalent programs. Such considerations are impossible in the Savage setting, where acts are functions.

Besides exploring these avenues of research, we would like to understand better the connection between our work and the work on *predictive state representations* (Jaeger 2000; Littman, Sutton, & Singh 2001). A predictive state representation is a way of representing dynamical systems that tries to move away from a state space as part of the problem description but, rather, constructs the state space in terms of tests and actions (which are taken to be primitive). While the technical details and motivation are quite different from our work, there are clearly similarities in spirit. It would be interesting to see if these similarities go deeper.

References

Anscombe, F. J., and Aumann, R. J. 1963. A definition of subjective probability. *Annals of Mathematical Statistics* 34:199–205.

Aumann, R. J. 1976. Agreeing to disagree. *Annals of Statistics* 4(6):1236–1239.

Fagin, R., and Halpern, J. Y. 1988. Belief, awareness, and limited reasoning. *Artificial Intelligence* 34:39–76.

Feinberg, Y. 2004. Subjective reasoning—games with unawareness. Technical Report Resarch Paper Series #1875, Stanford Graduate School of Business.

Feinberg, Y. 2005. Games with incomplete awareness. Technical Report Resarch Paper Series #1894, Stanford Graduate School of Business.

Gilboa, I., and Schmeidler, D. 1989. Maxmin expected utility with a non-unique prior. *Journal of Mathematical Economics* 18:141–153.

Gilboa, I., and Schmeidler, D. 2001. *A Theory of Case-Based Decisions*. Cambridge, Mass.: MIT Press.

Halpern, J. Y., and Rêgo, L. C. 2005. Interactive unawareness revisited. In *Theoretical Aspects of Rationality and Knowledge: Proc. Tenth Conference (TARK 2005)*, 78–91.

Halpern, J. Y., and Rêgo, L. C. 2006. Extensive games with possibly unaware players. In *Proc. Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*.

Heifetz, A.; Meier, M.; and Schipper, B. 2003. Multi-person unawareness. In *Theoretical Aspects of Rationality and Knowledge: Proc. Ninth Conference (TARK 2003)*, 148–158. An extended version with the title "Interactive unawareness" will appear in the *Journal of Economic Theory*.

Jaeger, M. 2000. Observable operator models for discrete stochastic time series. *Neural Computation* 12:1371–1398.

Kahneman, D.; Slovic, P.; and Tversky, A., eds. 1982. *Judgment Under Uncertainty: Heuristics and Biases*. Cambridge/New York: Cambridge University Press.

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

Littman, M. L.; Sutton, R. S.; and Singh, S. 2001. Predictive representations of state. In *Advances in Neural Information Processing Systems 14 (NIPS 2001)*, 817–823.

Modica, S., and Rustichini, A. 1999. Unawareness and partitional information structures. *Games and Economic Behavior* 27(2):265–298.

Savage, L. J. 1954. *Foundations of Statistics*. New York: Wiley.

Schmeidler, D. 1989. Subjective probability and expected utility without additivity. *Econometrica* 57:571–587.