A million answers to twenty questions: choosing by checklist*

Michael Mandler — Royal Holloway, University of London

Paola Manzini — University of St Andrews and IZA

Marco Mariotti† — University of St Andrews

This version: May 2010

Abstract

Several decision models in marketing science and psychology assume that a consumer chooses by proceeding sequentially through a checklist of desirable properties. These models are contrasted to the utility maximization model of rationality in economics. We show on the contrary that the two approaches are nearly equivalent. Moreover, the number of preference discriminations that an agent can make increases exponentially in the number of properties in the agent's checklist. Checklists therefore provide a rapid procedural basis for utility maximization.

JEL codes: D01

Keywords: Bounded rationality, utility maximization, choice function, lexicographic utility.

*We are grateful to Claire Blackman, Vincent Crawford, Peter Hammond, Jacob Leshno, Marco Li Calzi, Mark Machina, Ben Polak, Ludovic Renou, Ariel Rubinstein and Chris Tyson for comments and discussions.

†Mandler: Dept. of Economics, Royal Holloway College, University of London, Egham, Surrey TW20 0EX, UK. Manzini and Mariotti: School of Economics & Finance, Castlecliffe, The Scores, University of St Andrews KY16 9AL, UK.

1 Introduction

You go to a used car lot. You first state your maximum price, then ask if any cars with a manual transmission are available, then if any sport cars are available, then any Italian sport cars ... and you end up driving away in a red Alfa Romeo.

In this example you make your decision when facing a set of alternatives using only *properties* of the alternatives. A property is simply a subset of alternatives, e.g., all sports cars. You go through your checklist of properties until you are able to narrow down the set sufficiently. At each step you eliminate the alternatives that do not have the specified property, or, if no alternative has the property, you do not eliminate any options and move on to the next property. No maximization of utility or of preferences is invoked: all that is required is an ordered list of desirable attributes. That the list is ordered means that earlier properties always trump later properties; if the car buyer checks car color only with his final property, then color can never take precedence over the properties checked earlier on. This lexicographic feature of ordered properties makes choosing by checklist appear distant from the classical economic agent's pursuit of utility. Moreover, a checklist is easy to execute, while maximizing utility may seem to be a daunting task. In the words of Herbert Simon [23]:

> The assumption of a utility function postulates a consistency of human choice that is not always evidenced in reality. The assumption of maximization may also place a heavy (often unbearable) computational burden on the decision maker. (p. 16)

Checklists present a challenge to Simon's view. Although easy to use, checklists implicitly impose a utility ordering on alternatives; the checklist and utility models are in fact nearly equivalent. Checklists in addition can make fine preference discriminations using only a handful of properties; from the checklist point of view, utility maximization is computationally undemanding.

The sequential elimination of alternatives by whether or not they possess properties underlies several decision making models in psychology[1] and marketing science.[2] The specific checklist model we present is a simplified (deterministic) version of Tversky's [24] Elimination By Aspects, which

> is relatively easy to apply... involves no numerical computations and... is easy to explain and justify (p. 489 in [25]).

Observe that any decision procedure that follows a flowchart of 'yes or no' questions can be written as a checklist. Checklists can also serve as normative guides in fields such as clinical medicine. For example, Fischer et al. [7] propose a simple rule to guide the prescription of a certain antibiotic to treat pneumonia in young children. Because resistance can develop, this drug should be prescribed only in specific cases. The rule is (1) if the patient has had fever for less than two days, do not prescribe, (2) otherwise, and if the patient is less than three years old, do not prescribe, and (3) otherwise, prescribe. We will translate the car and antibiotic examples into the language of our model in section 2, where we incorporate 'deal-killing' properties that an option must possess in order to be chosen.

Decision-making with a checklist is considered basic precisely because it eschews any use of preference relations over alternatives, the hallmark of economic analysis. Its attraction is its simplicity: in the language of Gigerenzer and Todd [11], it generates 'fast and frugal' heuristics, appropriate when time, knowledge and computational power are scarce. Gigerenzer and Todd indeed emphasize the contrast between such heuristics and 'demonic rationality', by which they mean preference or utility maximization.

As the views of Simon and the psychologists illustrate, it is not clear at first sight that there is a connection between checklists and the economic model of maximization. And the fact that discriminations among alternatives made by one property can never be overturned by later properties suggests that the only maximizing agents that the model can capture are lexicographic agents who do not make trade-offs among different types of goods (where, e.g., agents prefer more of good 1 and good 2 quantities are decisive only when good 1 quantities are tied).

[1] E.g. from the classic 'elimination by aspect' model by Tversky [24], to the more recent Bereby-Meyer, Assor and Katz [1], Brandstätter, Gigerenzer and Hertwig [2] and Katsikopoulos and Martignon [14].

[2] See e.g. Yee et al. [26]. The term 'non-compensatory choice models' is used in these fields to underscore the lack of 'tradeoffs' between earlier and later properties.

We will see that the reverse is the case: the agents who use our benchmark model of checklists – where all the alternatives that will be rejected are eliminated in finitely many steps – always make choices that maximize some utility function. Since lexicographic preferences cannot be represented by a utility function, it follows that checklist users cannot be lexicographic. So, whatever goes on in the minds of checklist users, they act like classical maximizers. While we can extend the benchmark model of checklists to cover agents without utility functions, such agents remain handicapped: their checklists will go on indefinitely eliminating options, and never stop.

The lexicography example illustrates the broader principle, contra Simon, that having a utility function contributes to rather than detracts from decision-making efficiency. We will see that checklist users can sift through alternatives rapidly: the number of properties they must go through relative to the number of preference discriminations n that they make shrinks to 0 as n increases. Checklist users can in effect perform a binary search, which makes the number of preference discriminations they make an exponential function of the number of properties that they use. As a result, an agent who makes a 1,000,000 preference discriminations needs a checklist that is just 20 properties long.

Comparable conclusions hold for the agents of consumer theory who choose commodity bundles. It might seem that checklist users cannot exhibit the uncountably many indifference classes that textbook consumers have. But in fact the choice behavior of any utility-maximizer can be generated by some checklist. Moreover the checklist can be one of the benchmark checklists that execute quickly: for any finite set of alternatives, the agent will need to go through only finitely many properties on his or her checklist before coming to a decision.

So not only will any agent who uses our benchmark model of quickly executing checklists have a utility function but the converse holds as well: any utility-maximizer can make decisions with a quickly-executing checklist. The tractability that has attracted psychologists to checklists thus obtains if and only if checklist users display the trade-offs of utility maximizers.

We end up near the Gigerenzer and Todd [11] point of view but with a caveat. Checklists are indeed 'fast and frugal': they are a fast and frugal way to maximize utility.

2 Checklists

Fix a nonempty set of alternatives X. An agent faces a domain $\mathcal{A}$ of choice sets, where each A in $\mathcal{A}$ is a nonempty subset of X. For each choice set A in $\mathcal{A}$, the agent selects a nonempty $c(A) \subset A$. Following tradition, we call c a 'choice function' but each $c(A)$ is a set.

A decision maker who chooses by checklist decides on a $c(A)$ by going through a sequence of properties; for each property, if there is an alternative in A that has that property then the agent eliminates all those alternatives that do not. While an agent may use a large pool of properties to discriminate among alternatives, in our benchmark model we require that for every A a final selection is reached in a finite number of steps.

Formally, a *property* $P(i)$ is simply a set of alternatives, $P(i) \subset X$, and we say 'alternative x has property $P(i)$' when $x \in P(i)$. A *checklist* is a sequence of properties $P = (P(1), P(2), ...) = P(i)_{i \in I}$ where the set of indices I is either $\{1, ..., n\}$ or the entire set of natural numbers $\{1, ..., n, ...\}$.

Given a choice set $A \subset X$ and a checklist P, define inductively the following 'survivor sets' $S_i(A)$:

$$S_0(A) = A$$

$$S_i(A) = \begin{cases} S_{i-1}(A) \cap P(i) & \text{if } S_{i-1}(A) \cap P(i) \neq \varnothing \\ S_{i-1}(A) & \text{otherwise} \end{cases}$$

This sequence makes precise the elimination procedure we described. At each step i the agent checks whether the current set of surviving alternatives have the ith property. If some alternatives do, the alternatives that do not are thrown away. Otherwise, all alternatives survive to the next round. In both cases the agent moves to step $i + 1$.

Definition 1 *A choice function c defined on a domain $\mathcal{A}$* ***has a checklist*** *if and only if there exists a checklist P such that, for all $A \in \mathcal{A}$, there is a property $P(j)$ such that*

$$\begin{aligned} S_i(A) &= S_j(A) \text{ for all } i \geq j \\ c(A) &= S_j(A), \end{aligned} \tag{1}$$

and we then say that P is a checklist for c.

A choice function that has a checklist thus satisfies two features. First, the procedure 'finitely terminates': for any choice set A there exists a property in the checklist such that, from that stage onwards, the set of survivors does not shrink any further.[3] Second, this set of permanent survivors coincides with what the choice function selects from A.

The following two examples illustrate the versatility of the model.

Example 1 In the car example of the introduction, we can model the option of not choosing any car by letting some or all of the attributes be 'deal killers,' i.e., attributes that a car must have for a purchase to go through. For any car lot, let an object of choice be either a vehicle v_i in the lot, or the option w of walking away without buying anything. A choice set A (a car lot) then has the form $\{v_1, v_2, ..., v_n, w\}$. For the consumer in the introduction, with an ordered set of desirable attributes, the first s attributes will be deal killers if each of these properties includes w. For example, if attribute 1, say having price less than \$30,000, and attribute 2, having a manual transmission, are deal killers then $w \in P(1)$ and $w \in P(2)$ and then a A that has no manual transmission car cheaper than \$30,000 will lead the consumer to walk. If every attribute is a deal killer, let w be in each $P(i)$ and add an extra property that repeats the final $P(i)$ but omits w. Then if there is a car in A with every desirable attribute it is chosen, and w is eliminated by the extra property; otherwise, every car in A is eliminated and w survives as the only option. ■

Example 2 In the medical example in the introduction, a doctor faces a child who has had a fever for f days and whose age is y. A choice set A is thus a pair $\{(f, y, a), (f, y, na)\}$ indicating respectively that the child does or does not receive the antibiotic ((f, y, na) is similar to 'walking away' in Example 1). The doctor first makes sure that a child who has had a fever for less than two days does not receive the drug, which is accomplished with the property $P(1) = \{(f, y, na) : f < 2\}$. The doctor can then exclude children younger

[3] After reaching $P(j)$ in Definition 1, to execute a decision the agent must conclude that it would be pointless to consider any further properties. The agent can make this inference in two prominent cases: if S_j is a singleton or if S_j is a subset of a single indifference class (taking preferences as primitive in the latter case). The remaining cases are more problematic and 'finite termination' must be understood as an approximate description, as we will explain in section 7.

than 3 from treatment with the property $P(2) = \{(f, y, na) : y < 3\}$. If no option has been eliminated from A in the first two stages then the child has indeed had a fever for at least two days and is over three years old and a final property $P(3) = \{(f, y, a)\}$ will eliminate the option (f, y, na). There are checklists with fewer but more complex properties that deliver the same decisions but the simple properties above match the procedure that has been recommended to doctors. ■

3 Checklist users are utility-maximizers

Consider again the car buyer that opened the paper and the lexicographic flavor of his decisions. This agent makes a series of categorical judgments, where any discrimination between cars made by a property trumps all discriminations made by properties later in the checklist. It might therefore seem that the only preference relation that such a checklist user could maximize is lexicographic. Recall that lexicographic preferences, on $\mathbb{R}^2_+$ for example, are defined by $x \succsim y$ if and only if $x_1 > y_1$ or ($x_1 = y_1$ and $x_2 \geq y_2$): more of good 1 always trumps any increase of good 2.

The following example shows how to express categorical judgments as properties. We use the following terminology: we call a complete and transitive binary relation on X a *preference relation* and say that a choice function c on the domain $\mathcal{A}$ *maximizes a preference relation* $\succsim$ if $c(A) = \{x \in A : x \succsim y \text{ for all } y \in A\}$ for all $A \in \mathcal{A}$. A choice function c on the domain $\mathcal{A}$ *maximizes a utility function* if there exists a function $u : X \to \mathbb{R}$ such that $c(A) = \{x \in A : u(x) \geq u(y) \text{ for all } y \in A\}$ for all $A \in \mathcal{A}$. Maximizing a utility is a stronger condition than maximizing a preference: any c that maximizes a utility function automatically maximizes a preference relation but not vice versa, as the example of lexicographic preferences shows.

Example 3 (characteristics) In the spirit of Lancaster [16], we can recast the car example by viewing each car as a bundle of characteristics (horsepower, color, price, and so on). For any continuous characteristic, such as horsepower or price, there is a class of properties that we call 'coordinate cutoffs.' Suppose that there are two continuous characteristics and so $X = \mathbb{R}^2_+$. A coordinate cutoff is a property of the form $\{(x_1, x_2) : x_j \geq r\}$ or

$\{(x_1, x_2) : x_j \leq r\}$ where j is the coordinate 1 or 2 and r is a real number. Coordinate cutoffs express categorical judgments about a single characteristic, e.g., if coordinate 1 is price and $P(1) = \{(x_1, x_2) : x_1 \leq 30{,}000\}$ then any car costing less than \$30,000 is ranked above any other car. For instance, suppose an agent has four coordinate cutoffs all of the form $\{(x_1, x_2) : x_j \geq r\}$ with cutoff levels as given in Figure 1 (the cutoff of property $P(i)$ is labeled $r(i)$). The choice function with this checklist maximizes a utility function: in Figure 1 the regions from worst to best are labeled 1 through 9.

Properties in $\mathbb{R}^n_+$ of course do not have to be coordinate cutoffs. As a quick example, let coordinates 1 and 2 be two foods that make up a meal – say meat and potatoes. Then a property that placed a 800 calorie limit on the meal would be the 'calorie cutoff' $P(i) = \{(x_1, x_2) : k_1 x_1 + k_2 x_2 \leq 800\}$, where k_i is the number of calories per unit of food i. ■

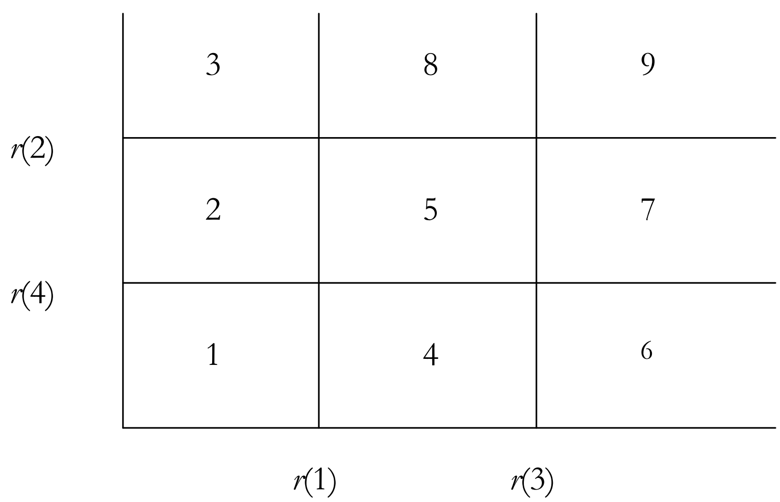

Figure 1: Coordinate cutoff preferences

The fact that the coordinate cutoff agent in Example 3 maximizes a utility function is no fluke and is not due to the simplification that the agent deploys only finitely many properties.

Theorem 1 *If a choice function has a checklist then it maximizes a utility function.*

Thus an agent whose choices come from a checklist acts 'as if' he is maximizing a utility function. Of course the agent does not have to think about preferences or utility at all; the agent may just be churning through his list of properties.

All proofs are in the appendix, but the argument behind Theorem 1 is simple. When a choice function has a checklist we can identify each $x \in X$ with a sequence of 'ins' and 'outs' that indicate in any coordinate i whether x is in or is not in property $P(i)$. Suppose we write down the 'ins' and 'outs' as a sequence of 1's and 0's respectively. When P has finitely many properties, use 0's following the last property. For example, if P has four properties and $x \in P(1)$, $x \notin P(2)$, $x \notin P(3)$, $x \in P(4)$, the sequence for x is $1, 0, 0, 1, 0, 0, 0, \ldots.$ Now we can read this sequence as the 0's and 1's of binary expansion of a number between 0 and 1; for the x above, this number is $\frac{1}{2} + \frac{1}{16} = .5625$. So, given a checklist P, each x has a sequence that defines a number $u(x)$ in the interval $[0, 1]$. Outside of a small class of exceptions, $u(x)$ can serve as the utility of x for an agent who uses P! The reason is simply that $u(x) \geq u(y)$ if and only if, at the first digit where the sequences for x and y differ, x has a 1 and y has a 0 and therefore in the first property i that contains one of x and y but not both it is x that must be in $P(i)$.[4]

Since lexicographic preferences cannot be represented by a utility function, we conclude from Theorem 1 that an agent who chooses with a checklist cannot have lexicographic preferences (even when the checklist consists of infinitely many coordinate cutoffs). Checklist users, who at first glance seem not to make trade-offs, turn out to fit the textbook description of an economic consumer.

In the next section, we will put a finer point on the problematic feature of lexicographic choice behavior. As we will see, it is easy to extend the checklist model to cover such behavior. Rather the problem is that lexicographic decision-making can be produced only by checklists that fail to terminate in finitely many steps and are therefore unwieldy. As we argued in the introduction, the absence of a utility function detracts from decision-making efficiency.

Theorem 1 leaves some important questions unanswered. While lexicographic behavior cannot arise from the standard checklists of this section, what types of choice behavior can? That of all utility maximizers or just certain types? In particular, could the utility-maximizing choices of textbook consumers arise from standard checklists? Since such agents

[4]The exceptions are numbers in $[0, 1]$ with two binary representations, but this difficulty can be bypassed by reading the sequence of 0's and 1's as a number written in base 3 (see the proof of Theorem 1).

have uncountably many indifference classes, it might seem that they are in the same boat as lexicographic agents and that their choices could not be the outcome of the practical checklists that finitely terminate. This turns out not be the case, but the following example lays out the potential for trouble.

Example 4 Suppose an agent uses the checklist $P(1)$, $P(2)$, ..., $P(n)$ where the properties form a partition of X (each $x \in X$ is in exactly one property). It is easy to see that this agent also maximizes a preference relation $\succsim$ with n indifference classes, $P(1), ..., P(n)$ going from best to worst: given a choice set A, no eliminations occur until the property $P(i)$ that contains the elements of A that have the lowest property index and at that stage all other elements of A are eliminated. Notice that we could omit the last property $P(n)$ without changing the choice function that results. ■

Example 4 makes clear that the choices of any utility maximizer with finitely many indifference classes could be the outcome of a checklist. But the construction in Example 4 uses the same number of properties as the number of indifference classes in $\succsim$ (or one fewer). The decisions made by the classical agents of consumer theory, who have uncountably many indifference classes, therefore could not arise from such checklists: these agents have too many indifference classes and if unsatiated have no top indifference class to begin the checklist.

The same 'slowness' feature of Example 4 also raises the possibility that even agents who make finitely many discriminations can be inefficient decision-makers who proceed through a large number of properties. We address these questions in sections 5 and 6.

4 Extended checklists and preference maximization

We now present a more abstract version of the model, allowing sequences of properties that go beyond the ordinary counting numbers; the new version opens the door to checklists for arbitrary preference-maximizing behavior including the lexicographic cases that failed to have the standard checklists of section 3. This section is more technical; since we will not refer back to these ideas until section 8, it can be skipped.

In our earlier elimination procedure, each set of survivors $S_i(A)$ is a subset of its immediate predecessor $S_{i-1}(A)$. Since therefore $S_{i-1}(A) = \bigcap_{k<i} S_k(A)$, we could equivalently define the elimination by

$$S_0(A) = A$$

$$S_i(A) = \begin{cases} \bigcap_{k<i} S_k(A) \cap P(i) & \text{if } \bigcap_{k<i} S_k(A) \cap P(i) \neq \varnothing \\ \bigcap_{k<i} S_k(A) & \text{otherwise} \end{cases}$$

for each $i > 0$. The new definition has the advantage that it can be applied to 'longer' sets of properties: we can weaken the assumption that the indices I in a checklist are a set of natural numbers and suppose instead that I is well-ordered by some $\leq$, letting 0 be the least element of I.[5] The assumption that I is well-ordered implies that each $i \in I$ has an immediate successor; thus the procession through the checklist of properties remains orderly. For an arbitrary well-ordered I, the above definition employs a variant of standard induction (transfinite induction) to specify each $S_i(A)$ as a function of its entire set of predecessors and $P(i)$.

We say that a choice function c has an **extended checklist** if c satisfies Definition 1 except that the $S_i(A)$ are defined as above and I is permitted to be any well-ordered set whose least element is 0.[6] The terminal step j continues to be defined as in Definition 1 but now need not be finite. Any of our earlier checklists, which we call 'standard,' qualifies as an extended checklist, and conversely, if c has an extended checklist that 'finitely terminates' – for each $A \in \mathcal{A}$, the index j identified in Definition 1 is finite – then c has a standard checklist since then we can excise all but the properties with finite indices.

The main advantage of extended checklists is that they give an exact characterization of preference maximization.

[5] A set B is *well-ordered* by $\leq$ if $\leq$ is a linear order (a complete, transitive, and antisymmetric relation) on B such that every nonempty subset of B has a least element b: $b \leq x$ for all $x \in B$. See Halmos [12] for the set theory concepts we use in this section.

[6] In terms of ordinal numbers, the distinction between standard and extended checklists is that the ordinal number of the former must be ω or less while the ordinal number of the latter is unrestricted. Also, notice that if we apply an arbitrary well-ordered set of properties to a choice set A, it could happen that $S_i(A)$ is empty for some i (when $\bigcap_{k<i} S_k(A) = \varnothing$). But if c has an extended checklist then this possibility does not arise since we require $c(A) \neq \varnothing$ for $A \in \Sigma$.

Theorem 2 *A choice function has an extended checklist if and only if it maximizes a preference relation.*

Since 'having an extended checklist,' as an assumption on choice functions, is equivalent to preference maximization, it is equivalent to any characterization of preference maximization for choice functions. For example, it is equivalent to the Richter [18] version of the strong axiom of revealed preference. Notice also that Theorem 2 imposes no restriction on the domain of the choice function; for example, it applies equally to budget sets in consumer theory and to finite sets.

Regarding the 'only if' half of Theorem 2, it is easy to detail the preference relation that an agent with an extended checklist implicitly maximizes. Recall from section 3, that when a choice function has a checklist, we can identify each x with the sequence of 'ins' and 'outs' that indicate in any coordinate i whether x is in or is not in property $P(i)$. We then declare $x \succsim y$ if the x and y sequences are identical or x scores an 'in' at the first coordinate where the sequences differ. Now if a eliminates b in the checklist's sequential eliminations – that is, if both a and b have survived to some stage $i-1$ but only a survives to stage i – then $a \succ b$ since the first property that has only one of a and b must have a and not b. So if x is $\succsim$-maximizing on some A then x could never be eliminated, and conversely if x is chosen from some A then $x \succ y$ must obtain for every y in A that is not chosen. So the choice function indeed maximizes the $\succsim$ we have defined.[7]

For the 'if' half of Theorem 2, suppose we are given a choice function c that maximizes a preference relation $\succsim$. We can then build an extended checklist from a familiar item, the better-than (weak upper-contour) sets of the preference relation $\succsim$: for each $x \in X$, set a property P_x equal to $\{y \in X : y \succsim x\}$, ignoring the duplicates that arise when $x \sim x'$. We then list – technically, we well-order – these properties to form an extended checklist. When this checklist is applied to some A, the agent will eventually hit a property P_x where $x \succsim y$ for all $y \in A$, whereupon no further eliminations occur.

[7] A less general argument works via the weak axiom of revealed preference (WARP). A choice function with an extended checklist must satisfy WARP since if x is chosen when y is available it must be that if there is a first property $P(i)$ that contains either x or y but not both then $P(i)$ contains x, hence if y is chosen from any S that contains x then x must be chosen too. So on any domain where WARP implies that a choice function maximizes some preference relation, for example the finite subsets of X, a choice function with a checklist must also maximize a preference relation.

Given Theorem 1, the checklists that produce lexicographic choices must be extended rather than standard (and the example below shows what they look like). Hence the problem that lexicographic behavior presents is not that it cannot arise from a checklist but only from checklists that are procedurally problematic. Utility-maximization in consumer theory, even though it typically involves uncountably many utility levels, does not suffer the same difficulty, but so far the only checklists we have seen that can generate such behavior (in the 'if' half of Theorem 2) are extended and hence need not finitely terminate.[8] To show that tractable checklists are consistent with the behavior of classical consumers, in section 6 we find replacement checklists that are standard.

Example 5 (characteristics revisited) To eliminate the puzzle of checklists and lexicographic preferences, suppose all of the properties in Example 3 are coordinate cutoffs of the form $\{(x_1, x_2) : x_j \geq r\}$, as in Figure 1. So if, e.g., $P(1) = \{(x_1, x_2) : x_1 \geq r\}$ then any bundle with $x_1 \geq r$ is ranked above any bundle with $x_1 < r$ according to the preferences maximized by any c that has $P(1)$ as its first property. If we increase the number of properties and let the cutoff levels 'fill in' each axis (become dense in $\mathbb{R}_+$), we approach preferences that have a strictly increasing utility function. As Theorem 1 showed, no matter how many coordinate cutoff properties an agent uses in a standard checklist, the preferences that result cannot lexicographically rank bundles first according the level of coordinate 1 and second according the level of coordinate 2. But there is an extended checklist for such preferences that uses coordinate cutoffs: begin with a countably infinite set of properties of type $\{(x_1, x_2) : x_1 \geq r\}$ (where the r's for these properties are dense in $\mathbb{R}_+$) and then proceed to a countably infinite set of properties of type $\{(x_1, x_2) : x_2 \geq r\}$ (again with the r's dense in $\mathbb{R}_+$).

5 Quick checklists 1: agents who finitely discriminate

Since checklists make for practical decision-making procedures only when the elimination of options concludes after finitely many steps, we need to pin down which preferences can arise

[8] The problem shows up in the proof of Theorem 2 when we take the nonconstructive step of well-ordering the upper contour sets to create the extended checklist.

from our benchmark model of 'standard' checklists. (Until further notice, all checklists will now be standard rather than extended.) Although finite termination might seem incompatible with preferences with uncountably many indifference classes, we will see that any case of utility-maximizing decision-making can be the outcome of a checklist that finitely terminates.

As Example 4 made clear, choice behavior that marks out a finite number of discriminations can always be the consequence of a checklist that finitely terminates. But finite termination is then too weak a test of practicality. Recall that the checklists in Example 4 lay out a worst-case scenario where the number of properties equals the number of preference discriminations. An agent that uses such a checklist would spend a long time eliminating alternatives before coming to a decision, ending up with a procedure that is plodding and profligate instead of fast and frugal.

This section and the next address these points. Can a checklist make a given finite number of discriminations reasonably quickly? And can the agents of consumer theory use a checklist at all?

Our measure of decision-making speed is the number of properties per preference discrimination. While a checklist user could of course use a single property and thus make decisions rapidly, such an agent would be dividing the universe of alternatives X coarsely, into just two indifference classes. So instead we ask how the potential number of preference discriminations n varies as a function of the number of properties that an agent uses or, equivalently, how many properties are needed to make n discriminations.

We define a checklist P to *make n discriminations* if there is a choice function c that maximizes a preference relation with n indifference classes and P is a checklist for c. To ensure that there are not multiple values for n that meet this definition, we assume in this section that the domain of choice sets includes every two-element subset of X.

To see how many properties are needed for a checklist to make n discriminations, let's label the indifference classes $1, ..., n$, going from worst to best. Since the number of properties required depends only on the number of indifference classes, we may as well take each indifference class to be a singleton. So our question becomes, for $X = \{1, ..., n\}$, how many properties are needed so that the choice function that results divides X into n indifference classes? Luckily the answer given by the checklists in Example 4 fails to be the minimum when $n > 1$.

Consider, as an example,

$$X = \{1, 2, 3, 4\}.$$

Given our labeling convention, let the choice function c maximize the usual order $\geq$ on integers. It is easy to see that $P(1) = \{4, 3\}$, $P(2) = \{4, 2\}$ is a checklist for c.

Next, consider

$$X = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

with c again maximizing $\geq$. Define the checklist $P(1) = \{8, 7, 6, 5\}$, $P(2) = \{8, 7, 4, 3\}$, $P(3) = \{8, 6, 4, 2\}$. Again, it is easy to verify that this is a checklist for c. (It suffices to consider just the two-element subsets of X.)

Notice how the first example is nested in the second: the last two properties $P(2)$ and $P(3)$ of the second example treat $\{5, 6, 7, 8\}$ and $\{1, 2, 3, 4\}$ just as P in the first example treats $\{1, 2, 3, 4\}$, with the additional first property $P(1)$ serving only to separate the two chains. So, we have provided a checklist with two properties makes four discriminations, and a checklist with just one additional property that makes twice as many discriminations. This conclusion extends inductively:

Theorem 3 *If X contains at least n alternatives then there is a checklist that makes n discriminations with k properties, where k is the smallest integer such that $2^k \geq n$. Furthermore, any checklist that makes n discriminations must have at least k properties.*

Theorem 3 shows how checklists become more and more efficient as the number of preference discriminations increases: the maximum number of preference discriminations n is an exponential function of the checklist length k. Or, if we take n as primitive, then the minimum number of properties required is a less-than-polynomial function of n and hence the ratio of the minimum number of properties to n falls to zero as n increases. Since $2^{20} \geq$ 1,000,000, Theorem 3 explains why a million preference discriminations require only twenty checklist properties.

We can compare the efficiency of a checklist to other choice procedures that make the same number of discriminations. Suppose an agent with n indifference classes wants to find the highest indifference class in a choice set; in the notation of the above examples, the agents seeks out, given $A \subset \{1, ..., n\}$, the largest integer in A. The solution of this problem via 'yes or no' questions is a classic illustration of binary search: first ask 'does A contain an integer between $\frac{n}{2}$ and n?', and then, if yes, ask 'does A contain an integer between $\frac{3n}{4}$ and n?' and, if no, ask 'does A contain an integer between $\frac{n}{4}$ and $\frac{n}{2}$?', and so on. That a recursive computer program, where the choice of the ith question depends on earlier answers, can execute this algorithm in $\lceil \log_2 n \rceil$ steps is hardly news ($\lceil x \rceil$ denotes the least integer $\geq x$).[9] What is notable about a checklist is that it executes the algorithm nonrecursively. A property $P(i)$ does not change as a function of the eliminations that occur prior to i, and every property is used for every A. To do without input from earlier steps, each property in effect encodes a set of questions. Consider again $X = \{1, 2, 3, 4, 5, 6, 7, 8\}$ and let m denote $\max A$. Then $P(1)$ 'asks' one question, 'is $m \in \{8, 7, 6, 5\}$?', $P(2)$ 'asks' two conditional questions, 'if $m \in \{8, 7, 6, 5\}$ then is $m \in \{8, 7\}$?' and 'if $m \notin \{8, 7, 6, 5\}$ then is $m \in \{4, 3\}$?, and $P(3)$ 'asks' four conditional questions. For $i > 1$, the eliminations prior to i ensure that only one of the antecedents of the $P(i)$ questions is satisfied. Property $P(i)$ therefore asks the right question, and without recursive instructions or an exhaustive tree of $n - 1$ 'if then' commands (where each answer to a command leads to a distinct subsequent command).

An optimal tree of 'yes or no' questions can in principle outperform a checklist. Suppose we can ask questions of the form 'does A intersect $Y \subset \{1, ..., n\}$?'. Then, depending on the probabilities that particular integers lie in A, the minimum expected number of questions can be less than $\lceil \log_2 n \rceil$. For example if it is highly likely that $m = \max A = 4$, then one can first ask 'does A intersect $\{5, 6, 7, 8\}$?' and if not 'does A intersect $\{4\}$?'. But if each $x \in X$ is equally likely to be m then $\lceil \log_2 n \rceil$ is the minimum expected number of questions: the optimal tree does no better than the optimal checklist.[10]

[9] See, e.g., Knuth [15], chapter 6, Theorem B.

[10] If questions of the form 'is $m \in Y$?' are permitted, which is exactly the game 'Twenty questions,' Huffman coding [13] generates the optimal tree. See also Zimmerman [27], and Gilbert [10] for the connection to our problem.

6 Quick checklists 2: classical utility maximizers

Checklists with a finite number of properties are appealingly concrete: there is a *uniform* upper bound on the number of properties the decision maker has to examine before the choice procedure terminates. When checklists are not restricted to be finite (but are still standard rather than extended), it remains true that each choice set needs to be checked against only finitely many properties but there might not be any bound on the number of properties that serves simultaneously for *all* choice sets. This small difference gives checklists much greater reach when they are not required to be finite. Indeed, we will now see that, subject to a domain restriction, for any case of utility-maximizing choice behavior there is always a checklist that generates that behavior. For utility functions with uncountably many utility levels, such as those found in classical consumer theory, these checklists are 'quick' in that the number of properties the agent must check is a negligible fraction of the infinite number of preference discriminations that the agent implicitly makes.

Given a choice function c that maximizes a utility function u, we may define a checklist P by setting, for each rational number r, the property $P_r = \{x \in X : u(x) \geq r\}$ and then listing these properties in any order. This checklist has a natural procedural interpretation: each 'r' represents a numerical satisfaction threshold. At each stage, the consumer keeps only the alternatives that are 'satisficing', if any, and otherwise he keeps all of them. Then in the next stage he modifies the satisfaction threshold. The u-maximal alternatives in a A will never be eliminated: if at any stage i the set of survivors from the previous rounds contains some alternatives that are in $P(i)$, then the u-maximal alternatives must be among them. Conversely, any z in A that is not u-maximal will eventually be eliminated by a P_r such that r lies strictly between $u(z)$ and the maximum utility achieved by the alternatives in A. Thus, we have

Theorem 4 *If a choice function defined on a domain of finite sets maximizes a utility function then it has a checklist.*

Theorem 4 shows the reach of checklists. They can generate utility-maximizing behavior that divides the universe of alternatives into a continuum of indifference classes and yet still (given the restricted domain) eliminate the inferior alternatives from a choice set in a finite number of steps.

The following example shows that a domain restriction is required in Theorem 4.

Example 6 Let X be the interval $[0,1]$, let the domain of c be the closed sets in X, let the utility function $u : X \to \mathbb{R}$ that c maximizes be defined by $u(x) = x$, and suppose P is a checklist for c. Proposition 1 below shows we may assume that the checklist consists only of properties $P(i)$ that are weak or strict upper contour sets, i.e., sets of the form $\{x \in X : x \geq q\}$ or $\{x \in X : x > q\}$ for some $q \in X$. That is, if $\widehat{P}$ is a checklist for c then there is also a checklist P for c that consists solely of upper contour sets.

Assume then that there is a P that is a checklist for c that consists of upper contours. If we call $\text{glb}(i)$ the greatest lower bound of $P(i)$, then there will be at most countably many $\text{glb}(i)$ for the properties in P. Pick some $y \in X$ that is not one of these $\text{glb}(i)$, and set $A = \{x \in X : x \leq y\}$. Then, for any i, $S_i(A)$ will equal the nonempty interval whose lower boundary equals $\max\{\text{glb}(k) : \text{glb}(k) < y \text{ and } k \leq i\}$ and whose upper boundary equals y. (This interval contains y but may or may not contain its lower boundary.) Since $S_i(A) \neq \{y\} = c(A)$ for all i, P could not in fact be a checklist for c. ∎

That we may take a checklist in Example 6 to consist solely of (weak or strict) upper contours illustrates a wider principle. A set $U \subset X$ is an *upper cut of a preference relation* $\succsim$ *on* X if $(x \in U \text{ and } y \succsim x) \Rightarrow y \in U$. For the preference relation $\geq$ on $\mathbb{R}$ (but not for an arbitrary preference relation), an upper cut must be a weak or strict upper contour set.

Proposition 1 (canonical checklists) *If c has a checklist and maximizes the preference relation $\succsim$, then c also has a checklist that consists solely of upper cuts of $\succsim$.*

If the domain of a choice function c is restricted sufficiently, c can maximize more than one preference relation; Proposition 1 applies to any these preference relations.

While Example 6 shows that some domain limitation is needed in Theorem 4, the restriction can be weakened. For instance, the conclusion of the theorem still holds on any domain that includes at most countably many infinite sets. But we do not have an attractive characterization of the maximum permissible domain. So, while the converse result, Theorem

1, is clearcut, the ideal way to fill the gap in 'A choice function ... if and only if it has a checklist' remains an open question.

7 Utility maximizers always have quick approximate checklists

As we have seen, the choice behavior of utility maximizers does not coincide exactly with that of agents who use a checklist (a domain restriction is necessary), nor of agents who use an extended checklist (since then we go beyond utility maximization to preference maximization). Nevertheless, checklists can closely *approximate* utility maximization regardless of the domain.

To capture the idea that a checklist can approximate the decision $c(A)$ we consider the limit of the set of survivors selected by a checklist: although the procedure never yields exactly the decision $c(A)$ at any finite step, it approximates $c(A)$ more and more accurately as the number of steps increases. In the limit, we get exact equivalence between the choices of checklist users and utility maximizers.

As no notion of distance is present in our set-up, we use a set-theoretic definition of the convergence of the $S_i(A)$. A choice function c on the domain $\mathcal{A}$ has an **approximate checklist** if and only if there is a checklist P such that, for all $A \in \mathcal{A}$,

$$c(A) = \bigcap_{i \in I} S_i(A),$$

where the $S_i(A)$ are defined from P as in section 2. Although after any finite number of steps the set of survivors may still contain other alternatives beside the chosen ones, it is only the chosen alternatives that survive all steps of elimination: for any alternative rejected by the choice function, there exists a property that it does not have.

Theorem 5 *A choice function maximizes a utility function if and only if it has an approximate checklist.*

Approximate checklists help explain how a checklist that has the entire set of natural numbers as its set of indices would work practically. Such checklists can raise a termination problem: even when no further eliminations occur after some property $P(j)$, the agent may not know this fact. The agent will know it for choice functions that always select singletons or subsets of a single indifference class (see footnote 3). But in all other cases, the practical distinction between ordinary and approximate checklists is not sharp. For both models, the agent would have to declare at some point that the set of alternatives has been winnowed down adequately.

8 Multivalued properties and the representation of preferences

While so far we have focussed on checklists as decision-making procedures, they can also be seen as a preference representation device. This section explores this possibility and the connection to Chipman [3]'s theory of lexicographic utility.

We can rephrase our initial model of standard checklists by replacing each property $P(i)$ with the indicator function of $P(i)$ – the function $u_i : X \to \{0, 1\}$ with $u_i(x) = 1$ if and only if $x \in P(i)$ – and redefining $S_i(A)$ to equal $\arg\max u_i(x)$ s.t. $x \in S_{i-1}(A)$ for all $i > 0$. Each of these newly defined $S_i(A)$ will coincide with our original definition of $S_i(A)$. For the more general case of extended checklists, we can instead use $S_i(A) = \arg\max u_i(x)$ s.t. $x \in \bigcap_{k<i} S_k(A)$ for $i > 0$.

This reformulation suggests replacing the u_i above with functions that have a larger range ('multivalued properties'). Among the prominent possibilities, we could admit any u_i that maps to a finite set with at least two elements, or any u_i that maps into $\mathbb{R}$. Indeed we could go one step further and instead of functions, use a complete and transitive relation R_i on X, and set

$$S_i(A) = \{x \in \bigcap_{k<i} S_k(A) : xR_iy \text{ for all } y \in \bigcap_{k<i} S_k(A)\} \tag{2}$$

for $i > 0$. This last proposal is evidently the most general. Given a well-ordered set of indices I with least element 0 and a complete and transitive R_i for each $i \in I$, we call $\{R_i\}_{i \in I}$ a *multivalued checklist*. If each R_i has at most two indifference classes (our original model) we say $\{R_i\}_{i\in I}$ is a two-valued checklist, if the number of indifference classes of each R_i is finite we say $\{R_i\}_{i\in I}$ is a finite-valued checklist, and if each R_i has a real-valued utility representation we say $\{R_i\}_{i\in I}$ is a real-valued checklist.

With the $S_i(A)$ given by (2), we can define $\{R_i\}_{i\in I}$ to be a multivalued checklist for a choice function c by applying Definition 1.

Theorem 2 extends to any multivalued checklist: a choice function c has a multivalued checklist if and only if it maximizes a preference relation. The 'if' direction follows from our original statement of Theorem 2. For the 'only if' direction, some minor adjustments to the proof of Theorem 2 show that if $\{R_i\}_{i\in I}$ is a multivalued checklist for c then c maximizes the weak lexicographic order $\geq_L$ on X defined by[11]

$$x \geq_L y \Leftrightarrow [(xR_iy \text{ and } yR_ix \text{ for all } i \in I) \text{ or } (yR_ix \Rightarrow \exists k < i \text{ with } xR_ky \text{ and } yR_kx)]. \quad (3)$$

Theorem 2's applicability to multivalued checklists suggests their use as a representation device. One way to proceed would be to say that a multivalued checklist $\{R_i\}_{i\in I}$ represents the preference relation $\succsim$ if $\{R_i\}_{i\in I}$ is a checklist for the choice function c, defined on finite subsets of X, that maximizes $\succsim$. But it is equivalent and simpler to omit any mention of choice functions and just say that a multivalued checklist $\{R_i\}_{i\in I}$ *represents the preference relation* $\succsim$ if $\succsim = \geq_L$ (as defined by (3)). Requiring that a checklist is n-valued (for $n =$ two, finite, real) provides a correspondingly more restrictive definition of representation.

A real-valued checklist is the definition of representation that Chipman [3] proposed in his classical work on utility theory.[12] To see that Chipman pitched his definition at the right level of generality, observe that with no restrictions on the admissible R_i, multivalued checklists can be trivial and have no value for representation purposes: any preference relation $\succsim$ can be represented by the multivalued checklist that consists of the single relation $\succsim$. Moreover, there are preference relations that can be 'concisely' represented by a real-valued checklist but that have neither a classical utility representation nor a 'concise' finite-valued checklist. The simplest example is the lexicographic ordering on $\mathbb{R}^2_+$, which can be represented by a

[11] This extension of Theorem 2 would not hold if the R_i were not required to be complete and transitive. See Manzini and Mariotti [17].

[12] We thank Chris Tyson for stressing the connection between our work and Chipman's. For a survey of Chipman's theory and related developments, see Fishburn [9].

real-valued checklist that consists of just two functions but where any finite-valued checklist representation must have an index set I that goes beyond $\mathbb{N}$ (this conclusion follows from Theorem 1). Thus real-valued checklists are restrictive enough to be useful but not so restrictive that they are always unwieldy. In fact, Chipman's construction would lose most of its value if we added even the smallest additional restriction on the admissible R_i, that each must have only countably many indifference classes: one may show that any such 'countably-valued' checklist that has an index set I that is finite or equal to $\mathbb{N}$ represents a preference relation that could also be represented by a classical utility function. To get a concise representation when a classical utility is unavailable, a real-valued checklist is required.

In our terminology, the main theorem in Chipman [3] states that any preference relation $\succsim$ can be represented by a real-valued checklist. Theorem 2 implies this result. Indeed, Chipman's proof uses utility functions with ranges that take on two values; thus, he implicitly showed that any $\succsim$ can be represented by a two-valued checklist, which is the content of Theorem 2.[13] Outside of Theorem 2, our results do not intersect with lexicographic utility theory, for the very reason that we restrict the range of the admissible R_i. The range restriction indeed exposes a rich structure hiding inside Chipman's theory; for example, the capacity of a two-valued checklist to make exponentially many preference discriminations has no parallel in the theory of real-valued checklists, since one real-valued function can by itself make infinitely many discriminations.

Finally, we note that our original model of two-valued checklists perform *reasonably* well as a representation tool. Chipman [4] showed that there are preferences relations that can be represented by only those real-valued checklists $\{R_i\}_{i \in I}$ that use a I that is uncountable. Since Theorem 2 applies to such preference relations, they can be represented by two-valued checklists – as Chipman himself makes clear – though of course I must again be uncountable. Conversely, if a preference relation $\succsim$ is represented by a real-valued checklist $\{R_i\}_{i \in I}$ with a set of indices I that is at most countable then it can be represented by a two-valued checklist where I is at most countable.[14] Real-valued checklists still have an edge: as we have seen,

[13] This result precedes Chipman in the mathematical literature on ordinal numbers, see Cuesta Dutari [5], [6] and Sierpinski [21].

[14] To build a two-valued checklist from such a $\{R_i\}_{i \in I}$, it is easiest to use our notation for properties.

there are preference relations $\succsim$ that can be represented by a real-valued checklist with a set of indices that is finite or equal to $\mathbb{N}$ but where the only two-valued checklists that represent the same $\succsim$ have to use a set of indices with an ordinal number larger than $\mathbb{N}$. Of course it is this 'drawback' of two-valued checklists that guarantees the tight connection between their tractability as a decision procedure – that they terminate after finitely many steps – and utility maximization. Two-valued checklists have to use a set of properties that goes beyond $\mathbb{N}$ to represent a $\succsim$ in just the cases where $\succsim$ has no utility representation.

9 Concluding remarks

Since Simon's [22] contribution, we have been used to thinking of 'procedural rationality' as entirely separate from, and even opposed to, 'substantive rationality.' This paper leads to a different view. We have considered a tractable, realistic procedure that can underpin utility maximization, blurring Simon's distinction. While this procedure is by no means the *only* possible one for a procedural agent, it is *a* tractable and realistic procedure in many contexts, as demonstrated by its popularity with psychologists.

There are ways to choose by checklist that do not fit the model of this paper. Consider a consumer shopping for a camera, who first looks for cameras on the top shelf, then for those priced between \$225 and \$250, and then for those with black finish. This agent could choose different cameras from stores that stocked the same set of cameras but put them on different shelves. Moreover, the properties (sets of cameras) in this list can differ by store whereas checklists as we have defined them are fixed across choice sets A. If we think of a store as a choice set, our model rules out this agent's choice procedure. Rubinstein and Salant [20] better fits this example: the alternatives in each choice problem are presented to the decision maker in an exogenously specified order (e.g., the element on the top shelf is seen before the element on the next shelf). A choice problem is then an ordered list of alternatives $(a_1, ..., a_k)$, and a choice function associates each such list with one of its elements.

For each R_i, there is an countable set $D_{R_i} \subset X$ that is R_i-order-dense. Hence for each R_i there is a function d_{R_i} that maps $\mathbb{N}$ onto D_{R_i} and we can define a property $P_{R_i}(j) = \{x \in X : x\, R_i\, d_{R_i}(j)\}$ for each 'index' (R_i, j) in the countable set $\{R_i\}_{i \in I} \times \mathbb{N}$. We define a well-ordering $\preceq$ of $\{R_i\}_{i \in I} \times \mathbb{N}$ by setting $(R_i, j) \preceq (R_m, l) \Leftrightarrow ((j \leq l \text{ and } i = m) \text{ or } i < m)$. It is easy to confirm that these properties as ordered by $\preceq$ define a two-valued checklist that represents $\succsim$.

There are also circumstances in which a checklist may not help much in simplifying choice. For example, while if you have to choose a type of fruit from {Cherries, Peaches, Apricots, Figs, Dates, Oranges, Apples, Pears} it may be natural to consider the properties 'Summer fruits' and 'Winter fruits', it is not so easy or natural to find properties to choose how to spend a retirement lump-sum among a disparate list such as {special holiday, new car, house makeover, golf life membership, live-in housekeeper, downpayment on kid's house, increase in pension pot, donation to charity}. So there is a sense in which utility maximization encapsulates a *specific* form of procedural rationality.

Although we believe that the checklist model is new to economics, we should mention Rubinstein [19], who underlines the potential importance of unary relations (what we call 'properties') in decision making. Although distantly related, that work was the initial stimulus for this project.

10 Appendix: Proofs

It is convenient to present the proofs of the first two results in the opposite order with respect to the main text.

Proof of Theorem 2: Let the choice function c have the extended checklist P. We identify each $x \in X$ with the vector $p_x \in \{0,1\}^I$ given by $p_x(i) = 1$ if $x \in P(i)$ and $p_x(i) = 0$ if $x \notin P(i)$ (of course each p_x can be associated with many alternatives). We order $\{0,1\}^I$ lexicographically: for $p, q \in \{0,1\}^I$, define $\geq_L$ by $p \geq_L q \Leftrightarrow (q(i) > p(i) \Rightarrow \exists k < i$ with $p(k) > q(k))$. The asymmetric and symmetric parts of $\geq_L$ are labeled $>_L$ and $=_L$ respectively. To conclude that $\geq_L$ is a linear order, we could appeal to the fact that the lexicographic order of any family of linear orders with well-ordered indices must itself be a linear order. But to argue directly, completeness follows from the fact that (1) if $p = q$ then $(q(i) > p(i) \Rightarrow \exists k < i$ with $p(k) > q(k))$ obtains vacuously, while (2) if $p \neq q$ then the well-ordering of I implies that $j = \min\{i : p(i) \neq q(i)\}$ is well-defined and hence $p >_L q$ if $p(j) > q(j)$ and $q >_L p$ if $q(j) > p(j)$. Case (2) also yields antisymmetry. For transitivity, if $p =_L q =_L r$ then $p = q = r$ and hence $p =_L r$. If on the other hand $p \geq_L q >_L r$ or $p >_L q \geq_L r$ set $j = \min\{i : p(i) \neq q(i) \text{ or } q(i) \neq r(i)\}$. Then $p(j) \geq q(j) \geq r(j)$ with at least one strict inequality. Hence $p(j) > r(j)$ and $p(i) = r(i)$ for $i < j$, i.e., $p >_L r$.

Let $\succsim$ now denote the relation on X given by $x \succsim y \Leftrightarrow p_x \geq_L p_y$: since $\geq_L$ on $\{0,1\}^I$ is a linear order, $\succsim$ on X is a preference relation. To see that for any $A \in \mathcal{A}$, $c(A) = \{x \in A : x \succsim y \text{ for all } y \in A\}$, suppose first that $x \in c(A)$. If $y \succ x$ for some $y \in A$ and we set $j = \min\{i : p_x(i) \neq p_y(i)\}$ then the fact that $x \in S_i(A)$ for all $i < j$ implies that $y \in S_i(A)$ for all $i < j$. But since $y \in P(j)$ and $x \notin P(j)$, $x \notin S_j(A)$, contradicting $x \in c(A)$. Conversely suppose $x \in A$ and $x \succsim y$ for all $y \in A$. Then, since $c(A)$ is nonempty, $x \succsim z$ for some $z \in c(A)$. Since $z \in S_i(A)$ for all i, $x \succsim z$ implies $\{i : p_x(i) \neq p_z(i)\} = \varnothing$ (otherwise z would be eliminated at $\min\{i : p_x(i) \neq p_z(i)\}$). So $x \in S_i(A)$ for all i, i.e., $x \in c(A)$.

Now suppose that c maximizes some preference relation $\succsim$. To construct a checklist, let $I = X \cup \{0\}$ and let $\leq$ be a well-ordering of I with $0 < x$ for any $x \in X$. (This is a nonconstructive step: the principle that any set can be well-ordered relies on the axiom of choice.) For each $x \in X$ define $P(x) = \{y \in X : y \succsim x\}$. Fix $A \in \mathcal{A}$ and some $x \in c(A)$. Then, for any $z \in X$ with $x \notin P(z)$, the fact that $x \succsim y$ for $y \in A$ and the transitivity of $\succsim$ imply $y \notin P(z)$ for any $y \in A$. So, for any $z \in X$, if $x \in \cap_{w<z} S_w(A)$ then $x \in S_z(A)$. Since $x \in S_0(A)$, transfinite induction implies that $x \in S_z(A)$ for all $z \in X$. Moreover, for all $y \notin c(A)$, $y \notin P(x)$ and so $y \notin S_x(A)$. Finally observe that $S_z(A) = S_x(A)$ for all z such that $x \leq z$, so that the terminal step j in Definition 1 is well defined. ■

Proof of Theorem 1: Let c have a checklist $P : I \to 2^X$. As in Theorem 2, given P, each $x \in X$ can be associated with a unique $p_x \in \{0,1\}^I$, where the i^{th} component is defined by $p_x(i) = 1$ if $x \in P(i)$ and $p_x(i) = 0$ if $x \notin P(i)$. Define $u : X \to \mathbb{R}$ by

$$u(x) = \sum_{i \in I} \frac{p_x(i)}{3^i}.$$

Since $\sum_{j>i} \frac{1}{3^j} < \frac{1}{3^i}$ for any $i \in I$, this u is a utility representation for $\succsim$, where, as in the proof of Theorem 2, $\succsim$ is the preference relation $\succsim$ on X induced by the lexicographic order $\geq_L$ on $\{0,1\}^I$. (A utility representation for $\succsim$ is a u such that $x \succsim y \Leftrightarrow u(x) \geq u(y)$.) The proof of Theorem 2 also shows that $c(A) = \{x \in X : x \succsim y \text{ for all } y \in X\}$ for all $A \in \mathcal{A}$. Hence $c(A) = \{x \in X : u(x) \geq u(y) \text{ for all } y \in X\}$. ■

Proof of Theorem 3: Given a P that makes n discriminations and the choice function $\widehat{c}$ for which P is a checklist, we may without loss of generality let $1, ..., n$ denote the indifference classes of the preference relation $\succsim$ that $\widehat{c}$ maximizes and let the linear order over $\{1, ..., n\}$ that c induces be $\geq$ (the standard order on the integers). That is, $g \geq h$ for $g, h \in \{1, ..., n\}$ if and only if, for all $x \in g$ and $y \in h$, $x \succsim y$. It is sufficient to consider only choice functions c defined on subsets of $\{1, ..., n\}$ that always selects the $\geq$-maximal element. Specifically, if $\widehat{c}$ is the choice function that maximizes $\succsim$, then let A be in the domain of c if and only if there is a $\widehat{A}$ in the domain of $\widehat{c}$ such that $\left((x \in \widehat{A} \text{ and } x \in g) \Rightarrow g \in A\right)$ and $\left(g \in A \Rightarrow (\exists x \in \widehat{A} \text{ such that } x \in g)\right)$.

Both conclusions of the theorem hold for $n = 1$ since the empty set of properties can then serve as the desired checklist. So assume henceforth that $n > 1$.

For the second half of the theorem, suppose c has a checklist P with s properties. As in the proof of Theorem 2, identify each $x \in \{1, ..., n\}$ with the $p_x \in \{0,1\}^s$ given by $p_x(i) = 1$ if $x \in P(i)$ and $p_x(i) = 0$ if $x \notin P(i)$. Since there are 2^s elements in $\{0,1\}^s$ and given that $n > 1$, $2^s < n$ would imply that $p_x = p_y$ for some distinct pair $x, y \in \{1, ..., n\}$. Since the domain of c contains the two-element sets, then $\{x, y\} \in \mathcal{A}$ and thus $c(\{x, y\}) = \{x, y\}$, contradicting the assumption that c maximizes $\geq$. So for this domain we cannot have $2^s < n$.

Regarding 'there is a checklist that makes n discriminations with k properties, where k is the smallest integer such that $2^k \geq n$,' suppose this claim holds for $1, ..., n-1$. Partition $\{1, ..., n\}$ into $Z_l = \{1, ..., m\}$ and $Z_u = \{m+1, ..., n\}$, where $m = n/2$ if n is even and $m = (n+1)/2$ if n is odd. Then, since $n > 1$, we have $2^{k-1} \geq |Z_r|$ for both $r = l$ and $r = u$. The induction hypothesis implies that $c|Z_u$ (the choice function defined by restricting c to subsets of Z_u) has a checklist $P = (P(1), ..., P(k-1))$ and that $c|Z_l$ has a checklist $P' = (P'(1), ..., P'(k-1))$. Define the checklist Q by $Q(1) = Z_u$ and $Q(i+1) = P(i) \cup P'(i)$ for $i = 1, ..., k-1$.

For any checklist R, let $S_i^R(A)$ denote the ith set of survivors when R is applied to the choice set A.

To see that Q is a checklist for c, notice first that if $A \in Z_u$ then $S_k^Q(A) = S_{k-1}^Q(A) = c|Z_u(A \cap Z_u) = c(A)$, and similarly if $A \in Z_l$ then $S_k^Q(A) = c(A)$. For all A that contain both elements of Z_l and elements of Z_u, application of $Q(1)$ yields $S_1^Q(A) = A \cap Q(1) = A \cap Z_u$. Since $Q(i+1) \cap Z_u = P(i)$, for $i = 1, ..., k-1$, application of properties $Q(2)$ through $Q(k)$

yields $S^Q_k(A) = S^P_{k-1}(A \cap Z_u) = c|Z_u(A \cap Z_u) = c(A)$. ■

Proof of Theorem 4: Given a choice function c that maximizes a utility function u, define for each rational r, the property $P_r = \{x \in X : u(x) \geq r\}$. We define a checklist P by letting $f : \mathbb{Q} \to \mathbb{N}$ be a bijection that enumerates the rationals and setting $P(i) = P_{f^{-1}(i)}$ for each $i \in \mathbb{N}$. Let A be a finite choice set. Then A has at least one u-maximal element, i.e., a $y \in A$ such that $u(y) \geq u(z)$ for all $z \in A$. Moreover, if y is u-maximal in A then the checklist P can never eliminate y: if $y \in S_{i-1}(A)$ and $z \in S_{i-1}(A) \cap P(i)$ then $y \in P(i)$ as well (since $u(y) \geq u(z)$) and hence y survives to stage i. If, on the other hand, $z \in A$ is not u-maximal then there is a u-maximal $y \in A$ and a property $P(f(r)) = P_r$ for some rational r such that $u(z) < r < u(y)$ and therefore z must be eliminated by property P_r if $z \in S_{i-1}(A)$. ■

Proof of Proposition 1: Let c with domain $\mathcal{A}$ have the (standard) checklist P and let I be the indices of P. We know from the proof of Theorem 2 that c maximizes the preference relation $\succsim$ on X defined by $w \succsim z \Leftrightarrow p_w \geq_L p_z$, where $p_w \in \{0,1\}^I$ is given by $p_w(k) = 1 \Leftrightarrow w \in P(k)$ and $\geq_L$ is the lexicographic order on $\{0,1\}^I$. Define the countable family $\mathcal{P}$ of upper cuts of $\succsim$ by $P_q = \{w \in X : p_w \geq_L q\} \in \mathcal{P}$ if and only if $q \in \{0,1\}^I$ has finitely many coordinates k such that $q(k) = 1$. Enumerate $\mathcal{P}$ by a bijection $\kappa : \mathcal{P} \to \widehat{I}$, where $\widehat{I}$ is $\mathbb{N}$ or $\{1, ..., n\}$, which defines a checklist $\widehat{P}$ and thus, for any $A \in \mathcal{A}$, a sequence of survivor sets $\widehat{S}_i(A)$. Since P is a checklist for c, for any $A \in \mathcal{A}$ there is an index $j \in I$ such that for any $y \in A \backslash c(A)$ there exists $i \in I$ with $y \notin S_i(A)$ and $i \leq j$; given any $y \in A \backslash c(A)$, let $i(y)$ denote the smallest such index i. Fix some $x \in c(A)$. Since each $\widehat{P}(k)$ is an upper cut of $\succsim$ and c maximizes $\succsim$, $x \in \widehat{S}_k(A)$ for all $k \in \widehat{I}$ (see the proof of Theorem 2). For any $y \in A \backslash c(A)$, we have $x \in P(i(y))$ and $y \notin P(i(y))$ while $x \in P(k) \Leftrightarrow y \in P(k)$ for $k < i(y)$. Thus

$$p_x \geq_L q^{i(y)} >_L p_y$$

where, for any index i, $q^i \in \{0,1\}^I$ is defined by $q^i(k) = x(k)$ for $k \leq i$ and $q^i(k) = 0$ for $k > i$. Thus, for the index $l = \kappa(P_{q^{i(y)}})$, $x \in \widehat{P}_l(A)$ and $y \notin \widehat{P}_l(A)$ and so $y \notin \widehat{S}_l(A)$. Since for any $y \in A \backslash c(A)$ the index $\kappa(P_{q^{i(y)}})$ must be drawn from the finite set $J = \{l \in \widehat{I} : l = \kappa(P_{q^i})$ for some $i \leq j\}$, $\widehat{S}_{\max J}(A) = c(A)$. Thus c has a checklist that consists of upper cuts of $\succsim$.

If in addition c maximizes the preference relation $\succsim'$, define the checklist P' by setting $P'(k) = \{z \in X : z \succsim' w \text{ for some } w \in \widehat{P}(k)\}$ for each $k \in \widehat{I}$, and survivor sets $S'_i(A)$. The transitivity of $\succsim'$ implies that, for any $k \in \widehat{I}$, $P'(k)$ is an upper cut of $\succsim'$. Consider some $A \in \mathcal{A}$ and suppose $x \in c(A)$ and $y \in A \backslash c(A)$. As in the previous paragraph, $x \in S'_k(A)$ for each $k \in \widehat{I}$. Moreover, $x \in P'(\kappa(P_{q^{i(y)}}))$ since $x \in \widehat{P}(\kappa(P_{q^{i(y)}}))$ and $\succsim'$ is reflexive, while $y \notin P'(\kappa(P_{q^{i(y)}}))$ since otherwise there would be a $w \in \widehat{P}(\kappa(P_{q^{i(y)}}))$ with $y \succsim' w$ and therefore, since $\widehat{P}(\kappa(P_{q^{i(y)}}))$ is an upper cut, $y \in \widehat{P}(\kappa(P_{q^{i(y)}}))$. Hence $y \notin S'_l(A)$ for $l = \kappa(P_{q^{i(y)}})$ and therefore $S'_{\max J}(A) = c(A)$. ■

Proof of Theorem 5: The part of the proof of Theorem 2 that shows that a c with a checklist $P : I \to 2^X$ maximizes the $\succsim$ induced by the lexicographic order on $\{0,1\}^I$ never uses the fact that P finitely terminates. The proof of Theorem 1 therefore also does not use finite termination, and so that proof establishes the 'if' part of the present Theorem. For the 'only if' part, where we are given a utility u that represents some $\succsim$ and a c that maximizes u, we use the same checklist constructed in the proof of Theorem 4. Once again for any $y \in c(A)$ and $i \in I$, we have $y \in S_i(A) \Rightarrow y \in S_{i+1}(A)$ and therefore $y \in S_i(A)$ for all $i \in I$. And for all $z \in A \backslash \{c(A)\}$, where therefore $y \succ z$ for any $y \in c(A)$, there must exist $P(i) = P_r$ such that $u(z) < r < u(y)$. So it must be that $z \notin \cap_{i \in I} S_i(A)$, and thus $c(A) = \cap_{i \in I} S_i(A)$. ■

References

[1] Bereby-Meyer, Y., A. Assor and I. Katz (2004) "Children's choice strategies: the effects of age and task demands," *Cognitive Development* 19: 127-146.

[2] Brandstätter, B., G. Gigerenzer, and R. Hertwig (2006) "The priority heuristic: making choices without trade-offs," *Psychological Review* 413: 409-432.

[3] Chipman, J. (1960) "The foundations of utility," *Econometrica* 28:193-224.

[4] Chipman, J. (1971) "On the lexicographic representation of preference orderings," in J. Chipman, L. Hurwicz, M. Richter, and H. Sonnenschein, eds., *Preferences, Utility, and Demand*. Harcourt: New York.

[5] Cuesta Dutari, N. (1943) "Teoria decimal de los tipos de orden," *Revista Matematica Hispano-Americana* 3: 186-205, 242-268.

[6] Cuesta Dutari, N. (1947) "Notas sobre unos trabajos de Sierpinski," *Revista Matematica Hispano-Americana* 7: 128-131.

[7] Fischer, J. E., F. Steiner, F. Zucol, C. Berger, L. Martignon, W. Bossart, M. Altwegg, and D. Nadal (2002) "Using simple heuristics to target macrolide prescription in children with community–acquired pneumonia," *Archives of Pediatrics* 156, 1005–1008.

[8] Fishburn, P. (1972) *Mathematics of Decision Theory.* Mouton: The Hague, Paris.

[9] Fishburn, P. (1974) "Lexicographic orders, utilities and decision rules: a survey," *Management Science* 20: 1442-1471.

[10] Gilbert, E. (1962) "Games of identification or convergence," *SIAM Review* 4: 16-24.

[11] Gigerenzer, G., P. M. Todd and the ABC Research Group (1999) *Simple Heuristics That Make Us Smart.* Oxford University Press, New York.

[12] Halmos, P.R. (1970) *Naive Set Theory.* Springer Verlag, New York.

[13] Huffman, D. (1952) "A method for the construction of minimum-redundancy codes," *Proceedings of the Institute of Radio Engineers* 40: 1098-1101.

[14] Katsikopoulos, K. V. and L. Martignon (2006) "Naive heuristics for paired comparisons: some results on their relative accuracy," *Journal of Mathematical Psychology* 50: 488-494.

[15] Knuth, D. (1973) *The Art of Computer Programming, Volume 3: Sorting and Searching.* Addison-Wesley: Reading MA.

[16] Lancaster, K.L. (1966) "A new approach to consumer theory", The *Journal of Political Economy*, 74: 132-157.

[17] Manzini, P. and M. Mariotti (2007) "Sequentially rationalizable choice," *American Economic Review* 97: 1824-1839.

[18] Richter, M. (1966) "Revealed preference theory," *Econometrica* 34: 635-645.

[19] Rubinstein, A. (1996) "Why are certain properties of binary relations relatively more common in natural language?" *Econometrica* 64: 343-355.

[20] Rubinstein, A. and Y. Salant (2006) "A model of choice from lists," *Theoretical Economics*, 1: 3-17.

[21] Sierpinski, W. (1949) "Sur une propriete des ensembles ordonnes," *Fundamenta Mathematicae* 36: 56-67.

[22] Simon, H.A. (1976) "From Substantive to Procedural Rationality," in S.J. Latsis, ed., Methods and Appraisal in Economics. Cambridge University Press, Cambridge.

[23] Simon, H.A. (1990) "Bounded Rationality," in J. Eatwell, M. MiIgate and P. Newman, eds., The New Palgrave: Utility and Probability. Macmillan, London.

[24] Tversky, A. (1972) "Elimination by Aspects: A Theory of Choice" *Psychological Review* 79: 281–299.

[25] Tversky, A. and E. Shafir (2004) *Preference, belief, and similarity: selected writings*, MIT Press.

[26] Yee, M., E. Dahan, J. R. Hauser and J. B. Orlin (2007) "Greedoid-based non-compensatory inference," *Marketing Science*, 26: 532-549.

[27] Zimmerman, S. (1959) "An optimal search procedure," *American Mathematical Monthly* 66: 690-693.