

Rational agents are the quickest*

Michael Mandler[†]

Royal Holloway College, University of London

This version: January 2009

Abstract

We consider the computational burden placed on agents who choose by proceeding through a checklist of criteria (for any pair of alternatives the first criterion that ranks the pair determines the agent's choice). An agent with complete and transitive preferences can use a 'quick' checklist where the number of criteria is a small proportion of the number of the agent's indifference classes. An irrational agent on the other hand can never use fewer criteria than a rational agent uses and on some domains must use more. Moreover, as the size of the domain increases, the proportion of irrational agents who can use a quick checklist goes to 0 at a super-exponential rate. In contrast to the view that utility-maximization is computationally demanding, it is irrational agents who face the more difficult procedural task.

JEL codes: D01

Keywords: Checklists, rationality, computational efficiency, lexicographic orders.

*I am grateful to Jose Apesteguia, Mordecai Kurz, Mark Machina, Paola Manzini, Marco Mariotti, Herbert Scarf, and particularly Claire Blackman for helpful discussions and advice. An extended abstract of this paper is published in the Proceedings of TARK XII and is available at <http://personal.rhul.ac.uk/uhte/035/Mandler.TARK.2009.pdf>

[†]Address: Department of Economics, Royal Holloway College, University of London, Egham, Surrey TW20 0EX, United Kingdom. Email: m.mandler@rhul.ac.uk

1 Introduction

Consider an agent who makes decisions by proceeding through a fixed sequence of criteria. For any pair of alternatives, the agent checks whether the first criterion recommends one of the alternatives over the other; if not, the agent proceeds to the second criterion; and so on. If no criterion ranks the alternatives, the agent selects both (declares either alternative to be acceptable). For example, the field of alternatives might be a set of possible meals, the first criterion might categorize meals by their inclusion of meat (vegetarian, red meat, white meat, all other cases), the second by country of origin (French, Italian, Chinese, all other countries), the third by flavor (predominantly salty, sweet, or bitter, or none of the above). Criteria need not be transitive – they might cycle for example – and they need not be complete. We call a sequence of criteria a *checklist*.

A checklist C will always implicitly maximize a preference relation \succ and we say C is a *checklist for \succ* . Conversely, given any preference \succ , there will under mild restrictions be a checklist for \succ . Our main question is computational: for which preferences \succ is there a *quick* checklist for \succ , that is, a checklist that employs the smallest theoretically possible number of criteria? To ensure the contest is fair, we compare preference relations that have the same number of indifference classes and endow criteria with the same capacity to discriminate among alternatives. We show that if \succ is complete and transitive – the preference is rational – then there is always a checklist for \succ that uses the theoretical minimum number of criteria; in fact, for rational preferences the ratio of the number of criteria in the quickest checklist to the number of indifference classes converges to 0 rapidly as the number of indifference classes increases. In contrast, *any* nonrational preference relation will fail to have a checklist that is as quick as those used by rational agents on some subset of the domain of alternatives and for some bound on the discriminatory power of the admissible criteria. Even a failure of completeness, which might seem to require fewer decision-making discriminations, will lead to a slow checklist on some domains.

We also calculate the proportion of irrational preferences on a given domain that have quick checklists, and show that as the size of the domain increases this proportion shrinks to 0 at a super-exponential rate. And finally we show that rational preferences can use

checklists that in expectation proceed through the smallest number of criteria.

This study pursues an agenda that Mandler, Manzini, and Mariotti (2008) (henceforth MMM) lays out but could not fully address. MMM considered checklists with binary criteria that partition alternatives into just two categories. The preferences that have checklists that use binary criteria are rational and conversely a rational preference always has a quick checklist that uses binary criteria. But because binary criteria do not allow an irrational preference relation to have any checklist at all, MMM could not stage a horse race between rational and irrational agents. Notice that each criterion in our opening ‘meals’ example divides meals into four types, not two; that example can therefore admit irrational preferences.

Our results contrast with the views of Herbert Simon (1976, 1990). Consider the following sample of Simon’s thinking:

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 (Simon (1990, p. 16)).

But if we measure computational burden by the number of checklist criteria, we come to a different conclusion: it is the irrational agents who bear the heaviest burden. Simon also seems to take utility to exist independently of an agent’s choices rather than being a mere representation of choice behavior. If instead we take an agent’s checklist as primitive then rational preferences (which can be represented by a utility function) could simply be the outcome of criteria that are themselves rational. In fact the most crude but also the most procedurally plausible criteria – binary criteria – are always rational and hence generate rational preferences. Agents might not use such criteria, but there is nothing computationally difficult about doing so. Indeed, the rational criteria (whether binary or not) have a capacity to make preference discriminations that outstrips any of the alternatives.

Tversky and Simonson (1993) adopt a view somewhere between Simon’s and ours; they recognize the dexterity of rational decision-making and are thus forced to find it paradoxical that agents often depart from rationality when they attempt to reduce decision-making

complexity.

Several other recent papers have considered sets or sequences of ‘criteria’ to represent choice behavior: Kalai, Rubinstein, and Spiegler (2002) (henceforth KRS), Apesteguia and Ballester (2005, 2008a, 2008b) (henceforth AB), and Manzini and Mariotti (2007). Besides MMM (2008), KRS and AB (2008b) are closest to the present paper in that they assess a set of criteria by the number of criteria in the set (what we call ‘length’). But KRS and AB pursue a different agenda; they seek the most concise explanations of an agent’s decisions, not the minimum speed with which an agent comes to a decision (on the latter see Salant (2003)). The number of equivalence classes used in criteria is therefore irrelevant, whereas for our aim of comparing the computational efficiency of different agents it is imperative to give every agent access to criteria of equivalent discriminatory power. Also, agents in our model do not use an unordered set of criteria; they proceed lexicographically through a fixed sequence of criteria, where it is the first criterion that ranks x and y that determines the checklist’s overall ranking of x and y , as in Manzini and Mariotti (2007), AB (2008a), MMM (2008), and much earlier the lexicographic utility theory of Chipman (1960, 1971) (see also Fishburn (1974)). Chipman’s framework in fact bears close resemblance to ours, but because Chipman was pursuing a representation for preferences that do not have a classical utility function, he allowed criteria to make a continuum of discriminations; our measure of computational efficiency therefore cannot be applied.

2 Criteria and Checklists

An agent will be represented by an asymmetric binary relation \succ , henceforth a *preference relation*, on a set of alternatives X , called the *domain of \succ* .

The agent makes decisions by proceeding through a sequence of criteria C_1, \dots, C_T . Each criterion C_i is also an asymmetric relation on X but criteria will typically divide X into a smaller number of equivalence classes than preference relations do. For a variation on the example in the introduction, X could be the set of all movies and an agent might chose between movie rentals by going through a list of criteria: one criterion

might categorize movies by the languages the agent speaks (movies in French, movies in English, movies in Spanish, all others), a second criterion might categorize by type of movie (thriller, comedy, science fiction, all others), and so on. Criteria need not be transitive and a criterion C_i need not rank every pair of the equivalence classes that C_i implicitly defines.

Definition 1 *The sequence $C = (C_1, \dots, C_T)$ is a checklist for \succ if and only if*

- (1) *each C_i is an asymmetric relation on X , and*
- (2) *$x \succ y \Leftrightarrow \exists i$ with $1 \leq i \leq T$ such that $x C_i y$ and not $y C_j x$ for all $j < i$.*

The integer T is the length of C .

We will also say that a preference \succ *has the checklist C* if C is a checklist for \succ . Definition 1 amounts to a mild generalization of Chipman's (1960) lexicographic definition of utility representation, and of kindred concepts in set theory, e.g., Cuesta Dutari (1943, 1947).

There is no need to take \succ as primitive in Definition 1. We may instead directly identify an agent with a checklist C ; an agent with C need not think about the preferences he or she is implicitly maximizing. Of course we may still deduce \succ from C using (2) above.

Example 1 Some seemingly crude decision procedures while clearly sequential may not at first appear to qualify as checklists. Suppose an agent chooses between two cars x and y by proceeding through a list of 'properties,' P_1, P_2, P_3, \dots , where each P_i is a subset of the domain of all cars X . The agent first check if x and y have property 1 (i.e., if $x \in P_1$ and $y \in P_1$). If one of the cars has property 1 but the other does not then the agent chooses the car that does. If both cars or if neither car has property 1 then the agent proceeds to property 2, and so on. Here is a sample of possible properties:

$$\begin{aligned}
 P_1 &= \{\text{all cars that cost less than \$25,000}\} \\
 P_2 &= \{\text{all four-door cars}\} \\
 P_3 &= \{\text{all Japanese cars}\} \\
 P_4 &= \{\text{all Toyotas}\} \\
 &\vdots
 \end{aligned}$$

So this agent first sifts by a price cut-off, then sifts by whether a car is a four-door, then by whether the car is made in Japan, and so on. This list of properties, an example of the model in Mandler, Manzini, Mariotti (MMM) (2008), qualifies as a checklist of criteria since the agent could make the same decisions by using a ‘binary’ criterion C_1 that ranks any car in P_1 as strictly superior to any car in $X \setminus P_1$, a binary C_2 that ranks any car in P_2 as strictly superior to any car in $X \setminus P_2$, and so on. ■

Whether we take the preference \succ or checklist C in Definition 1 as primitive, a checklist with a small number of criteria relative to the number of indifference classes in \succ means that the checklist C is making choice discriminations quickly. Our task will be to see which \succ are the quickest.

Checklists can evidently be seen as a decision procedure for choice sets A that consist of two elements. Given $A = \{x, y\}$, the agent proceeds through the checklist until he comes to the first criterion C_i that ranks x and y and then eliminates the C_i -inferior item; if no criterion ranks x and y the agent selects both alternatives. But checklists can also be applied to larger choice sets: if we feed an arbitrary choice set A to the checklist $C = (C_1, \dots, C_T)$ we eliminate in each round i any element of A that has survived so far and that C_i ranks as inferior to some other survivor. We thus define ‘survivor sets’ $S_i(A)$ recursively by

$$S_0(A) = A$$

$$S_i(A) = \{x \in S_{i-1}(A) : \nexists y \in S_{i-1}(A) \text{ with } y C_i x\} \text{ for } i = 1, \dots, T,$$

(as in Manzini and Mariotti (2007), Apestegua and Ballester (2008a), MMM (2008)). Suppose c is a choice function on a domain \mathcal{D} that includes the two-element sets; so $c(A)$ is a nonempty subset for any nonempty $A \in \mathcal{D}$. We then say that a choice function c ‘can use the checklist C ’ if $S_T(A) = c(A)$ for all $A \in \mathcal{D}$. While choice from arbitrary sets is the principal interpretation we have in mind for checklists, the extra generality of A ’s with more than two elements would be pointless for our purposes. As we will see, an agent with a rational preference \succ can use the shortest possible checklist when choosing from two-element sets; and it will be clear that a choice function defined on arbitrary

sets that maximizes \succ can use exactly the same checklist. Thus rational choice functions can use the shortest possible checklists. We will point out in section 4 that the converse that holds for preferences also holds for choice functions: *only* the choice functions that maximize a rational preference will be ‘uniformly quick’.

3 Indifference and equivalence classes

We measure the speed or efficiency of a checklist by the number of criteria it uses, that is, by the checklist’s length. The minimum length of a checklist for a \succ is determined in part by the number of indifference classes in \succ and in part by the number of discriminations that checklist criteria make. We need a definition of equivalence classes for asymmetric relations that gives an accurate count for both concepts.

Definition 2 *Given an asymmetric relation $>$ on X , the binary relation \approx on X is defined by*

$$x \approx y \Leftrightarrow \{z \in X : z > x\} = \{z \in X : z > y\} \text{ and } \{z \in X : x > z\} = \{z \in X : y > z\}$$

and is called the equivalence (or indifference) relation of $>$.

It is easy to confirm that \approx is in fact an equivalence relation and thus furnishes an appropriate definition of indifference for preferences. For a textbook discussion of \approx , specialized to the case where \succ is transitive, see Fishburn (1970).

Definition 3 *Given an asymmetric relation $>$ and its equivalence relation \approx , a $>$ -equivalence class is a nonempty $I \subset X$ such that (1) $x, y \in I \Rightarrow x \approx y$ and (2) $(x \in I \text{ and } x \approx y) \Rightarrow y \in I$.*

Given an asymmetric relation $>$ on X , the $>$ -equivalence classes form a partition of X .

When we apply Definitions 2 and 3 to a preference relation \succ we use \sim to denote the equivalence relation of \succ and call it an *indifference relation*. To keep our language tidy, we reserve *equivalence* for equivalence relations of criteria. We assume throughout the remainder of the paper, and without further remark, that any preference \succ has finitely many indifference classes.

To see how the count of criterion equivalence classes works, consider again the ‘meals’ example at the beginning of the paper. In that example, each criterion we listed happened to partition meals into four categories, e.g., into French, Italian, Chinese, and a remainder category of all other national cuisines. If a criterion C_i is a ranking of these categories then for any x and y in the same category $x C_i y$ does *not* obtain and further C_i must rank x and y the same way relative to any z in a different category. If in addition the C_i ranking assigns each category distinct upper and lower contour sets – for instance when C_i actually ranks each pair of categories – then C_i will have four equivalence classes. In Example 1, where criteria are defined from properties, each criterion specifies just two equivalence classes, those items in the property and those items excluded.

Turning to preferences, a word of explanation is needed on why we begin with a strict preference \succ and define indifference classes from \succ rather than taking a weak preference \succeq as primitive and letting the symmetric part of \succeq be our definition of indifference. The difficulty with the latter strategy is that a \succeq might generate two or more indifference classes, unranked by \succeq , that have identical upper and lower contour sets; that is, we might have x and y such that neither $x \succeq y$ nor $y \succeq x$ holds but where

$$\{z \in X : z \succ x\} = \{z \in X : z \succ y\} \text{ and } \{z \in X : x \succ z\} = \{z \in X : y \succ z\}$$

where \succ denotes the strict part of \succeq . Assuming that an agent with \succeq chooses all \succ -undominated elements from any choice set, x and y are fully interchangeable from a behavioral point of view. Hence no criterion in any checklist for \succeq would need to distinguish between x and y . So if we were to label x and y as being in separate indifference classes, we would give a checklist for \succeq unwarranted credit for tackling a \succeq with more indifference classes than show up in the agent’s behavior. Conversely, \succeq might classify two elements as indifferent when their upper and lower contour sets do differ and hence are behaviorally distinguishable.

Definition 4 *Given a positive integer p , $C = (C_1, \dots, C_T)$ is a p -checklist for \succ if and only if C is a checklist for \succ and each C_i has p or fewer C_i -equivalence classes.*

For convenience, we have assumed that each criterion C_i in a p -checklist has the same upper bound p on C_i -equivalence classes. See the discussion following Theorem 2 for the minor differences that arise when p varies by criterion.

It may well be that an irrational agent, overwhelmed by the difficulty of life, makes only a few preference discriminations or even none; such an agent, not surprisingly, can use a very short checklist. So, to compare like with like we run a race only between agents who share the same number of preference indifference classes and let the length of the shortest p -checklist that each can use be the test of who wins.

4 Rational versus irrational agents

We are almost set to run the speed competition between rational and irrational agents. To define rationality, we use a definition of completeness that applies to strict preference relations.

Definition 5 *A preference relation \succ on the domain X is complete if and only if for every $x, y \in X$ either $x (\succ \cup \sim) y$ or $y (\succ \cup \sim) x$ or both.*

Equivalently, one may define \succ to be complete if for all \succ -equivalence classes $I \neq J$, either $I \succ J$ or $J \succ I$.

We identify ‘rationality’ with a \succ that is complete and transitive. It is easy to confirm that an asymmetric relation \succ is complete and transitive if and only if it is negatively transitive (i.e., where $x \not\succeq y \not\succeq z$ implies $x \not\succeq z$), the traditional definition of rationality for asymmetric relations.

One way that a rational \succ can arise is when each criterion C_i in a p -checklist $C = (C_1, \dots, C_T)$ is itself rational (complete and transitive) on X . To see this, for any $x \in X$ and criterion i , let x_i be the number of C_i -equivalence classes J for which $x C_i J$,¹ and then identify x with the integer whose expansion in base p has x_i as its i th digit: call this integer $\text{Int}(x)$. If C is a checklist for \succ , then $x \succ y$ if and only if the first criterion C_i that ranks x and y has $x C_i y$, i.e., the first digit i where $\text{Int}(x)$ and $\text{Int}(y)$ differ in base p

¹For any binary relation R on X , any subsets $A, B \subset X$, and any $c \in X$, we use $A R B$ to mean $a R b$ for all $a \in A, b \in B$ and use $c R A$ to mean $c R a$ for all $a \in A$.

(reading from the left) has $x_i > y_i$ and so $\text{Int}(x) > \text{Int}(y)$. It is also easy to confirm that $x \sim y$ if and only if $\text{Int}(x) = \text{Int}(y)$. Since $\text{Int}(x)$ may therefore serve as a utility number for any $x \in X$, \succ has a utility representation and is therefore complete and transitive:

Observation 1 *If each criterion in a checklist C is complete and transitive and C is a checklist for \succ , then \succ is complete and transitive.*

So, if we think of the checklist rather than \succ as primitive, it is not ‘difficult’ for an agent to have rational preferences, as Simon suggests in his work; rational preferences can merely indicate that an agent uses a checklist with complete and transitive criteria. Indeed, the crudest criteria – those that arise from properties (Example 1) where $p = 2$ – are automatically complete and transitive. Hence the preferences that have 2-checklists are necessarily rational; properties look primitive, but they lead to choice behavior that Simon took to be demanding.

Since only rational agents can have 2-checklists a comparison of the checklist length of rational and irrational agents forces us to use higher values of p . The following observation shows that we need only go to $p \geq 3$.

Observation 2 *Any preference \succ has a p -checklist if $p \geq 3$.*

The simplest way to build a p -checklist for \succ , if $p \geq 3$, is just to list the preference rankings in \succ : for any two \succ -indifference classes I and J with $I \succ J$, define a criterion $C_{I,J}$ by $x C_{I,J} y$ if and only if $x \in I$ and $y \in J$, and then form a checklist C that consists for all such criteria, arranged in any order, and no others. Such a C is evidently a checklist for \succ and it is a p -checklist for any $p \geq 3$ since each $C_{I,J}$ has the three equivalence classes, namely I , J , and $X \setminus (I \cup J)$. An alternative and usually shorter construction is to define for each \succ -indifference class I a criterion C_I by $y C_I x$ if and only if $x \in I$ and $y \succ x$, and again let C consist of all such criteria, arranged in any order. We again get a p -checklist for \succ if $p \geq 3$, this time with a number of criteria equal to the number of \succ -indifference classes n . If \succ is complete, our first construction in contrast uses $\frac{n(n-1)}{2}$ criteria. The worst case – where the quickest checklist has length n – can in fact occur (see Example 3).

We turn to the lower bound on the length of the quickest checklist. For any real number a , let $\lceil a \rceil$ denote the smallest integer b such that $b \geq a$.

Theorem 1 *A preference relation \succ with n indifference classes has no p -checklist with length strictly less than $\lceil \log_p n \rceil$.*

Proof. If (C_1, \dots, C_T) is a p -checklist for \succ , we may define the p -checklist (C'_1, \dots, C'_T) for \succ by

$$x' C'_i y' \Leftrightarrow (\exists x, y \in X \text{ such that } x \sim x', y \sim y', \text{ and } x C_i y)$$

for $i = 1, \dots, T$. Then, for any $a, b \in X$ with $a \sim b$ and any $i \in \{1, \dots, T\}$, a and b are always in the same C'_i -equivalence class. Therefore, if we let $[x]$ denote the \succ -indifference class that contains x , the preference relation \triangleright on $\{I \subset X : I = [x] \text{ for some } x \in X\}$ defined by $[a] \triangleright [b]$ if and only if $a \succ b$ must also have a p -checklist of length T . It is therefore without loss of generality to assume that each \succ -indifference class is a singleton.

Given a p -checklist C of length k , define a C -undecided set to be a $A \subset X$ such that $x, y \in A \Rightarrow$ (for each $i = 1, \dots, k$, \exists a C_i -equivalence class I such that $\{x, y\} \in I$), define a *maximal C -undecided set* to be a C -undecided set A such there there does not exist a C -undecided set B with $|B| > |A|$, and finally define the *maximal C -undecided cardinality* to equal $|A|$ where A is a maximal C -undecided set.

We show, by induction on k , that if the p -checklist C has length k then the maximal C -undecided cardinality is greater than or equal to $p^{-k}n$. This claim holds for $k = 1$ since if every C_i -equivalence class has strictly fewer than $p^{-k}n$ elements we would have $|X| < n$. Assuming then that the claim holds for arbitrary k , let $C = (C_1, \dots, C_k, C_{k+1})$ be an arbitrary p -checklist of length $k + 1$. So there is a (C_1, \dots, C_k) -undecided set A with $|A| \geq p^{-k}n$. Since the p C_{k+1} -equivalence classes, I_1, \dots, I_p , form a partition of X , the sets $I_1 \cap A, \dots, I_p \cap A$ partition A ; hence one of these sets, say $I_j \cap A$, must have at least $p^{-(k+1)}n$ elements. And since $I_j \cap A$ is a $(C_1, \dots, C_k, C_{k+1})$ -undecided set, the maximal $(C_1, \dots, C_k, C_{k+1})$ -undecided cardinality is at least $p^{-(k+1)}n$.

If C is a p -checklist for \succ of length k , then the maximal C -undecided cardinality must equal 1. If it did not then there would be distinct $x, y \in X$ and a C_i -equivalence class I_i for each $i = 1, \dots, k$ such that $x, y \in I_i$ for $i = 1, \dots, k$; and then for all $z \in X$ there would

be a $R \in \{\succ, \prec\}$ such that $xRz \Leftrightarrow yRz$, i.e., x and y are in the same \succ -indifference class (here and subsequently \prec denotes $\{(a, b) \in X \times X : b \succ a\}$). Hence, the claim of the previous paragraph gives $p^{-k}n \leq 1$ or, equivalently, $\log_p n \leq k$. Since k is an integer, $\lceil \log_p n \rceil \leq k$. ■

Theorem 1 leads to the following definition of a quick checklist.

Definition 6 *Given a preference relation \succ with n indifference classes, call a p -checklist for \succ with length $\lceil \log_p n \rceil$ quick.*

The label ‘quick’ is well-deserved when Definition 6 applies. If we fix p , then, as the number of \succ -indifference classes n increases, $\log_p n$ increases at a less-than-polynomial rate.

For any complete and transitive \succ and any $p \geq 2$, \succ has a quick p -checklist. The proof of Theorem 2 shows this formally (and applies to the $p = 2$ case not covered in the statement of the theorem). But an example will be instructive.

Example 2 For $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, let \succ be the usual ordering of the integers. The unique quick 3-checklist is then the (C_1, C_2) given by

$$\begin{aligned} &\{9, 8, 7\} C_1 \{6, 5, 4\} C_1 \{3, 2, 1\} \text{ and } \{9, 8, 7\} C_1 \{3, 2, 1\} \\ &\{9, 6, 3\} C_2 \{8, 5, 2\} C_2 \{7, 4, 1\} \text{ and } \{9, 6, 3\} C_2 \{7, 4, 1\}, \end{aligned}$$

which is pictured in Figure 1. Criterion C_1 divides X into a top third, a middle third, and a bottom third, and then ranks these thirds as \succ does, while C_2 groups together the top items from the three C_1 -equivalence classes, the middle items from the three C_1 -equivalence classes, and the bottom three items of the three C_1 -equivalence classes, and then linearly ranks the tops over the middles over the bottoms. ■

Example 2 in effect reverse engineers the construction used for Observation 1, and provides a model for how to build a quick checklist for any complete and transitive \succ . One begins by partitioning X into p C_1 -equivalence classes that are ordered by \succ . Then at each subsequent stage i , for each maximal $A \subset X$ that has not yet been divided into separate criterion equivalence classes, partition A into p cells so that these cells are ordered

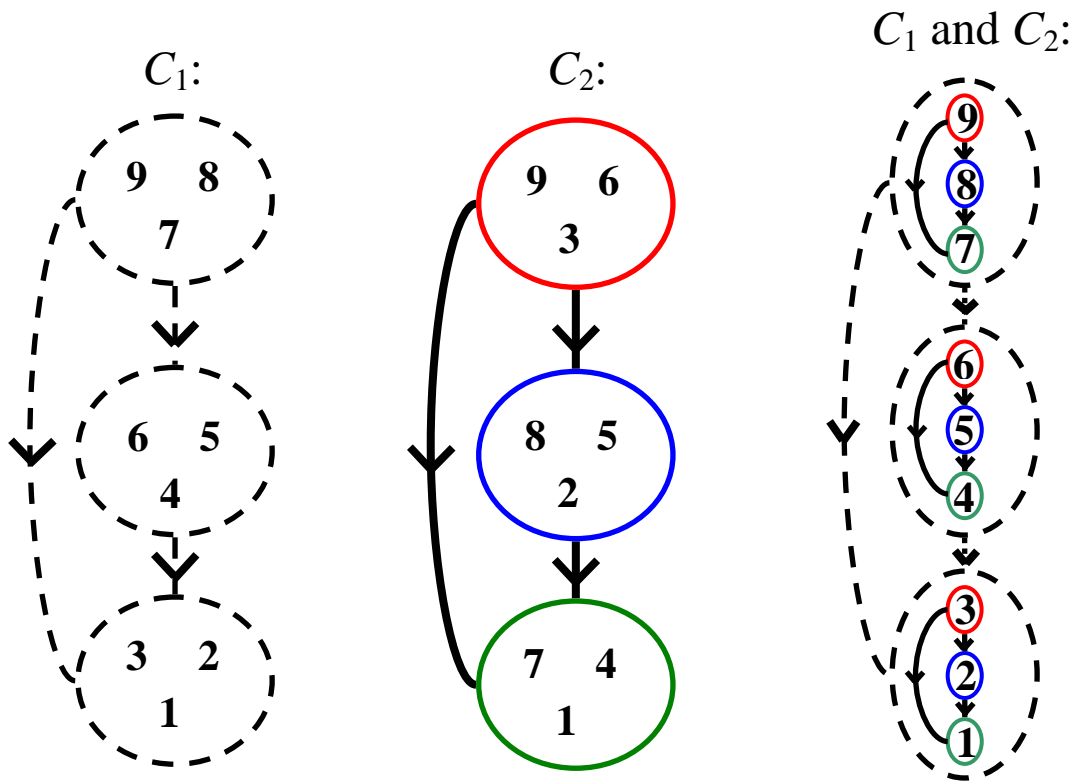


Figure 1: A quick 3-checklist for a rational \succ

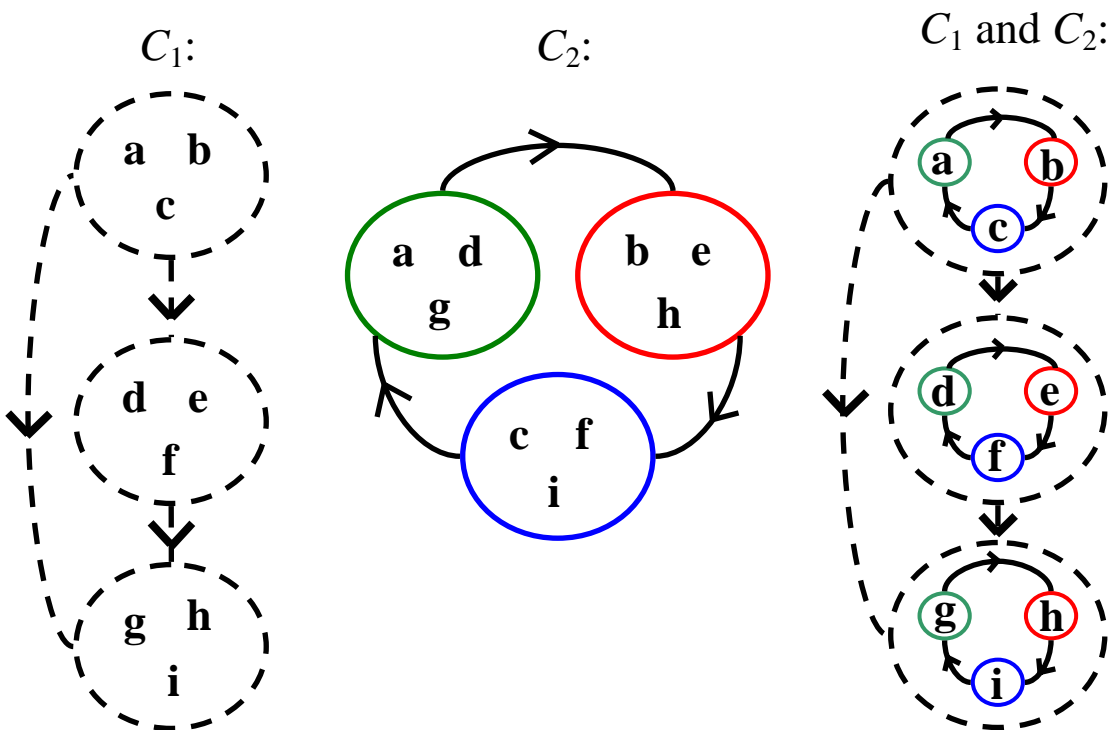


Figure 2: A quick 3-checklist for an irrational \succ

by \succ and group together all the cells across A 's that have the same rank according to \succ to form the C_i -equivalence classes. When the number of \succ -indifference classes n is an exact power of p , we must make sure that the cells at each stage contain the same number of \succ -indifference classes; in all other cases, there will be some latitude in how to proceed but one may always follow the rule that the cells into which each A is divided have sizes that differ by at most one \succ -indifference class. See the proofs of Theorems 2 and 5 for more details.

The key fact that ensures that this construction works – and the key to the decision-making speed of a rational \succ – is that each subset of alternatives A is ‘ordered the same way’ by \succ . Formally, for a rational \succ any two sets of \succ -indifference classes with the same cardinality will be order isomorphic and hence a criterion can order them both in a way that respects \succ .² Only rational preferences have such a rich supply of order-isomorphic subsets; in fact the order-isomorphism property we have stated fully characterizes the rational preferences with at least four indifference classes (see Mandler (2009)).

So one can build a quick checklist for a complete and transitive \succ from a sequence of complete and transitive criteria; conversely, Observation 1 reported that any sequence of complete and transitive criteria will generate a \succ that is complete and transitive. We will now see that a checklist user that implicitly maximizes a rational \succ can use a checklist that makes choice discriminations more efficiently than any other checklist user who shares the same p .

Consider the following \succ that has no quick checklist (and hence is not rational).

Example 3 For $X = \{a, b, c, d\}$, let \succ be defined by $a \succ b \succ c \succ d$ (so \succ is both incomplete and intransitive). Suppose $p = 3$. If there were a quick 3-checklist, it must have length $2 = \lceil \log_3 4 \rceil$. So then either C_1 makes two out of the three \succ -rankings – that is, C_1 has $a C_1 b C_1 c$ or $b C_1 c C_1 d$ or $(a C_1 b$ and $c C_1 d)$ – or C_1 makes one \succ -ranking and C_2 makes the other two. If the criterion C_i that makes two rankings has $a C_i b C_i c$ then the omitted item d must be placed in the same C_i -equivalence class as one of $\{a, b, c\}$ and thus be ranked by C_i vis-à-vis b if joined with a or c and be ranked vis-à-vis a if joined with b , contrary to the rankings given by \succ . The C_i that makes two rankings similarly

²Two sets, where each is endowed with a binary relation, (A, \geq_A) and (B, \geq_B) , are order-isomorphic if and only if there is a bijection $f : A \rightarrow B$ such that, for all $x, y \in A$, $x \geq_A y \Leftrightarrow f(x) \geq_B f(y)$.

cannot have $b C_i c C_i d$. And finally if C_i has $(a C_i b$ and $c C_i d)$ then (since $p = 3$) either $\{a, c\} C_i \{b, d\}$ or $a C_i \{b, c\} C_i \{d\}$, which again imposes rankings not given by \succ .

The shortest-length 3-checklist for \succ has length 3. Here are two samples:

$$C_1 = \{(a, b)\}, C_2 = \{(b, c)\}, C_3 = \{(c, d)\}$$

$$C'_1 = \{(a, b)\}, C'_2 = \{(b, c)\}, C'_3 = \{(c, d), (c, b)\}.$$

For C'_3 , the three C'_3 -equivalence classes are $\{a\}$, $\{b, d\}$, and $\{c\}$.

The above \succ is not quite the worst case. The \succ 's that come in last in the checklist-efficiency horse race are the \succ 's with n indifference classes that require a 3-checklist of length n . The smallest n where such a \succ arises is 5: let \succ on $\{a, b, c, d, e\}$ be defined the cycle $a \succ b \succ c \succ d \succ e \succ a$. The argument that there is no 3-checklist shorter than length 5 is similar to the case above and we leave the details to the reader (see also Mandler (2009)). ■

One moral of Example 3 is that while it may seem that incomplete preferences should have a speed advantage (one's criteria need to make fewer rankings), incompleteness forces an agent to make sure that any \succ -unranked pair is unranked by *every* criterion. This restriction burns up criterion equivalence classes and hence slows down the quickest checklist.

The plot thickens, however. In some carefully constructed cases, an irrational agent – an agent with a \succ that fails to be complete and transitive – can have a quick checklist.

Example 4 Let $X = \{a, b, c, d, e, f, g, h, i\}$ and define \succ by

$$\{a, b, c\} \succ \{d, e, f\} \succ \{g, h, i\} \text{ and } \{a, b, c\} \succ \{g, h, i\}$$

$$a \succ b \succ c \succ a \text{ and } d \succ e \succ f \succ d \text{ and } g \succ h \succ i \succ g.$$

Thus there are three triples that \succ orders linearly but within each triple \succ orders the items as a cycle. The following 3-checklist (C_1, C_2) for \succ , pictured in Figure 2, is quick:

$$\{a, b, c\} C_1 \{d, e, f\} C_1 \{g, h, i\} \text{ and } \{a, b, c\} C_1 \{g, h, i\}$$

$$\{a, d, g\} C_2 \{b, e, h\} C_2 \{c, f, i\} C_2 \{a, d, g\}.$$

Outside of the labeling the only difference between the present example and Example 2 is the C_2 is a cycle rather a line. In both cases the C_1 -equivalence classes, as ordered by \succ are order isomorphic, which is a necessary condition for a checklist to be quick. ■

We can mimic Example 4 to generate all of the \succ 's with nine indifference classes that have quick 3-checklists. For any nine-indifference-class \succ with a quick 3-checklist (C_1, C_2) , each C_i must have three C_i -equivalence classes and each C_i -equivalence class must consist of three \succ -indifference classes. Moreover, we must put the three \succ -indifference classes in each C_1 -equivalence class into separate C_2 -equivalence classes. A 3-checklist that does not meet these requirements would have to put some pair x, y such that not $x \sim y$ in *both* the same C_1 -equivalence class *and* the same C_2 -equivalence class, and then since x and y would be ranked the same way relative to every other $z \in X$ we would have $x \sim y$ (the proof of Theorem 1 makes this point in more detail). Now up to an order isomorphism there are only four asymmetric orderings of three equivalence classes. Labeling the equivalence classes a, b , and c , these are: (1) $\{(a, b), (b, c), (a, c)\}$, (2) $\{(a, b), (b, c), (c, a)\}$, (3) $\{(a, b), (b, c)\}$, and (4) $\{(a, b)\}$.³ By letting C_1 order the C_1 -equivalence classes according to one of the orderings (1) through (4) and letting C_2 order the items in each C_1 -equivalence class according to one of the orderings (1) through (4) (the same ordering for every C_1 -equivalence class), we can generate, up to an order isomorphism, all of the preferences with nine indifference classes that have quick 3-checklists.⁴ Evidently there are 16 such checklists: Example 4 is one (C_1 is the ordering (1) and C_2 is the ordering (2)) and Example 2 is another (both C_1 and C_2 are the ordering (1)). Now each of these 16 checklists is a checklist for one preference \succ (up to an order isomorphism) and 15 of these \succ are not rational. Since up to an order isomorphism there are approximately 400 billion distinct irrational \succ 's with nine indifference classes it is apparently an exceedingly unusual event for an irrational \succ with nine indifference classes to have a quick 3-checklist.

We further pursue the question of how many irrational \succ 's have quick checklists in section 5. Here we take a different line of attack and argue that for any irrational \succ there is always a subdomain of X on which \succ does not have a quick p -checklist for some

³The orderings $\{(a, b), (c, b)\}$, $\{(a, b), (a, c)\}$, and \emptyset do not qualify since they have 2, 2, and 1 equivalence classes respectively.

⁴We assume here that the \succ -indifference classes are singletons.

p. Notice the delicacy of the checklist in Example 4: to achieve quickness, the items in each of the three C_1 -equivalence classes must be ordered by \succ in the same way (i.e., be order isomorphic). Consequently, if we were to swap one of the items in X in the example for a tenth item j then in order for the new domain to have a quick 3-checklist the item j would have to be \succ -ordered relatively to the eight remaining items in the same way as the item that j replaces. For one particular swap, we may be lucky – j may have the desired fit – but if we in turn swap j for each of the original nine we will necessarily come to a case where j will fail to share the \succ -orderings of the omitted item. This fact would not be pertinent if \succ were complete and transitive since then we could build new checklist criteria according to the model laid out in Example 2 (recall our earlier observation that when \succ is rational, any pair of equinumerous sets of indifference classes will be order isomorphic). But, a little surprisingly, the swapping procedure we have laid out will generate a domain on which \succ has no quick checklist even if the source of the irrationality of \succ is merely incompleteness. Although the details are long and messy, this is a rough guide to how the proof of our main theorem (Theorem 2) proceeds. There is one proviso however: the arguments we have given for Theorem 2 do not apply if the largest possible domain for \succ contains just nine or fewer \succ -indifference classes. If this largest domain has, say, exactly nine indifference classes, then there may well be a quick 3-checklist C for an irrational \succ and it is easy to see that on any subdomain the checklist formed by dropping from C the ranking of elements that are missing in the subdomain will remain quick. So the smallest domain where it might be the case that an irrational \succ will fail to have a quick checklist on some subdomain must have at least 10 indifference classes. This is in fact the domain restriction we impose in Theorem 2; the irrational agents must fail to have a quick checklist on some subdomain at the first point where this failure is conceivable.

We lay out the definitions that formalize the above sketch.

Definition 7 *Let \succ be a preference relation with domain X . The binary relation \succ' is a subrelation of \succ if and only if there is a subdomain $X' \subset X$ such that $\succ' = \succ \cap (X' \times X')$.*

We may think of a subdomain X' in two ways. First, it could be that every so often – on the first of every month – the agent discovers that only alternatives in X' will be

available in the coming month. So during the month the agent need only use a checklist for the subrelation of \succ that has domain X' . Our question is then ‘which agents can use a quick checklist every month?’ Second, the agent or the world might set the domain X' on which the checklist operates once and for all but we as outside analysts do not know what this X' is. We then ask ‘which agents necessarily have the flexibility to use a quick checklist no matter how X' is set?’ Similarly, the number of discriminations p that criteria can make might also change periodically or we do not know the p that the agent will use (or be allowed to use). We therefore formally define what it means for an agent to be capable of using a quick checklist under all circumstances as follows.

Definition 8 *The preference \succ has uniformly quick checklists if and only if for all subrelations \succ' of \succ and all $p \geq 3$ there is a quick p -checklist for \succ' .*

Theorem 2 *A preference with $n \geq 10$ indifference classes has uniformly quick checklists if and only if it is complete and transitive.*

Extension to choice functions. In section 2, we saw that checklists define choice functions c that select from arbitrary choice sets. Set the domain of any c to include all of the two-element subsets of X . If c maximizes a rational preference \succ then c can use the quick checklist that the proof of Theorem 2 (or Example 2) employs. To get a converse, define the number of indifference classes generated by a choice function c to be the number of indifference classes in the base relation R of c^5 and say the c ‘can use uniformly quick checklists’ if for any $X' \subset X$ and any $p \geq 3$ the restriction of c to X' can use a p -checklist with $\lceil \log_p n' \rceil$ criteria, where n' is the number of indifference classes in the base relation of the same restriction of c . Theorem 2 then implies that c ‘can use uniformly quick checklists’ if and only if its base relation is rational. It is easily confirmed that the base relation of a choice function c that can use a checklist is rational if and only if c maximizes a rational preference. So a full equivalence obtains between rationality and uniformly quick choice functions.

Extension to variable capacity checklists. So far we have required each criterion C_i in a checklist to share a common upper bound p on C_i -equivalence classes. We could

⁵ R is defined by $aRb \Leftrightarrow (c(\{a, b\}) = \{a\})$.

instead specify a sequence of upper bounds $\vec{p} = (p_1, p_2, \dots)$ where each $p_i \geq 3$. Suppose \succ has n indifference classes and let $p(n)$ denote the smallest integer such that $\prod_{i=1}^{p(n)} p_i \geq n$. We can then call a \vec{p} -checklist for \succ that has $p(n)$ criteria *quick* and define a \succ to have *uniformly quick checklists* if and only if for all subrelations \succ' of \succ and all \vec{p} , there is a quick \vec{p} -checklist for \succ' . The proof of the ‘if’ half of Theorem 2 extends easily to these new definitions while the harder ‘only if’ half of the Theorem extends vacuously given that our previous definitions are a special case of the new definitions.

We give the proof of the ‘if’ half of Theorem 2 here and relegate the tedious proof of the ‘only if’ half to the appendix.

Half of proof of Theorem 2. Suppose \succ is complete and transitive. Fix a p and a subrelation \succ' of \succ with domain X' and with n indifference classes. For any $x \in X'$, let $[x]$ be the \succ' -indifference class that contains x . For any \succ' -indifference class $I \subset X'$, define

$$r(I) = |\{J \subset X : J \text{ is a } \succ' \text{-indifference class such that } I \succ' J\}|,$$

which can be viewed as the common utility number of the elements in I . Next, identify each integer r such that $0 \leq r \leq n - 1$ with its expansion in base p . That is, for each r there is a unique ordered sequence of $T = \lceil \log_p n \rceil$ integers, $(k(r, 1), \dots, k(r, T))$, where $0 \leq k(r, j) \leq p$ for $j = 1, \dots, T$, such that $r = \sum_{j=1}^T k(r, j)p^{T-j}$. For $j = 1, \dots, T$, define C_j by $x C_j y$ if and only if $k(r([x]), j) > k(r([y]), j)$. For any $x, y \in X'$, suppose there is a smallest integer i with $1 \leq i \leq T$ such that $k(r([x]), i) \neq k(r([y]), i)$. If say $k(r([x]), i) > k(r([y]), i)$ then $r[x] > r[y]$ and so $x \succ' y$. Moreover, $x C_i y$ and not $y C_k x$ for all integers $k < i$, as desired. If on the other hand $k(r([x]), i) = k(r([y]), i)$ for all integers i with $1 \leq i \leq T$, then $r([x]) = r([y])$ and so $x \sim' y$. And then not $x C_i y$ and not $y C_i x$ for all i with $1 \leq i \leq T$. ■

5 The proportion of irrational preferences with quick checklists

We fix $p \geq 3$ and consider the proportion of preferences that have quick p -checklists as the size of the domain increases. We first consider the proportion of all preferences with quick checklists – it is a more straightforward calculation – and then turn to our real concern, irrational preferences.

For any n , let X_n denote a generic domain of n objects, $\{1, \dots, n\}$, let $\succ(n)$ denote the set of all preferences on X_n ,

$$\succ(n) = \{\succ \subset X_n \times X_n : \succ \text{ is asymmetric}\},$$

and let $|\succ(n)|$ denote the cardinality of $\succ(n)$. As for quick checklists, let $q(n)$ denote the set of preferences on X_n that have quick checklists,

$$q(n) = \{\succ \in \succ(n) : \succ \text{ has a quick } p\text{-checklist}\},$$

and let $|q(n)|$ denote the cardinality of $q(n)$.

Theorem 3 *The proportion of preferences on X_n that have quick p -checklists converges to 0 at a super-exponential rate as $n \rightarrow \infty$. That is, for any $k > 0$, $e^{kn} \frac{|q(n)|}{|\succ(n)|}$ converges to 0.*

Proof. Since there are $\frac{n(n-1)}{2}$ unordered pairs of elements in X_n and since for any $x, y \in X_n$ an asymmetric relation \succ must specify one of three possibilities ($x \succ y$, $y \succ x$, or (not $x \succ y$ and not $y \succ x$)), $|\succ(n)| = 3^{\frac{n(n-1)}{2}}$.

The number of ways that X_n can be partitioned into p cells is given by the Stirling number of the second kind $S(n, p)$ (see Goldberg et al. (1972) and Knuth (1997) and for the formula for $S(n, p)$ we use below). The number of asymmetric binary relations on the domain $\{1, \dots, p\}$ is $3^{\frac{p(p-1)}{2}}$, a total that includes all of the asymmetric binary relations with m equivalence classes for any $m \leq p$. Hence $S(n, k)3^{\frac{p(p-1)}{2}}$ provides an upper bound on the number of asymmetric binary relations C on X_n with p or fewer C -equivalence

classes. Finally, since a preference on X_n has at most n indifference classes, a quick p -checklist has no more than $\lceil \log_p n \rceil$ criteria (Theorem 1). Hence an upper bound $U(n)$ on $|q(n)|$ is given by

$$U(n) = \left(S(n, k) 3^{\frac{p(p-1)}{2}} \right)^{\lceil \log_p n \rceil}.$$

Since (Goldberg et al. (1972)) we may write $S(n, p)$ as

$$S(n, p) = \frac{1}{p!} \sum_{i=0}^p (-1)^{p-i} \binom{p}{i} i^n,$$

we have

$$S(n, p) \leq \frac{1}{p!} \sum_{i=0}^p \binom{p}{i} p^n.$$

Hence there is a $r > 0$ such that $S(n, k) 3^{\frac{p(p-1)}{2}}$ is bounded above by r^n . Hence

$$U(n) \leq (r^n)^{1+\log_p n} = e^{(1+\log_p n)n \ln r}.$$

So, for any fixed $k > 0$, $e^{kn} |q(n)|$ is bounded above by $e^{(1+\log_p n)n \ln r + kn}$. Since $|\succ(n)| = 3^{\frac{n(n-1)}{2}} = e^{\frac{n(n-1)}{2} \ln 3}$ and since

$$\lim_{n \rightarrow \infty} \left(\frac{n(n-1)}{2} \ln 3 - ((1+\log_p n)n \ln r + kn) \right) = \infty,$$

$e^{kn} \frac{|q(n)|}{|\succ(n)|} \rightarrow 0$ as $n \rightarrow \infty$. ■

For all but the smallest values of n , the set of possible preferences on X_n is vast. Only a comparative handful of these are rational; when n is reasonably large, most preferences on X_n have n indifference classes, and yet there are only $n!$ preference relations on X_n with n indifference classes that are complete and transitive (and up to an order isomorphism there is in fact only one rational preference on X_n with n indifference classes). It is therefore no surprise that the conclusion of Theorem 3 continues to hold if we narrow our focus to irrational preferences.

Let $\succ_{\text{ir}}(n)$, $|\succ_{\text{ir}}(n)|$, $q_{\text{ir}}(n)$, and $|q_{\text{ir}}(n)|$ denote the restriction of $\succ(n)$, $|\succ(n)|$, $q(n)$, and $|q(n)|$ to preferences that are not complete and transitive (e.g., $\succ_{\text{ir}}(n) = \{\succ \in \succ(n) :$

\succ is not complete and transitive} and $q_{\text{ir}}(n) = \{\succ \in \succ_{\text{ir}}(n) : \succ \text{ has a quick } p\text{-checklist}\}$).

Theorem 4 *The proportion of irrational preferences on X_n that have quick p -checklists converges to 0 at a super-exponential rate as $n \rightarrow \infty$. That is, for any $k > 0$, $e^{kn} \frac{|q_{\text{ir}}(n)|}{|\succ_{\text{ir}}(n)|}$ converges to 0.*

Proof. Only one adjustment is needed to the previous proof. For each asymmetric binary relation \succ_{n-1} on X_{n-1} we may select a binary relation R on X_n that only ranks n relative to the elements of X_{n-1} (each ordered pair in R has n as one of its coordinates) such that $\succ_{n-1} \cup R$ is in $\succ_{\text{ir}}(n)$. Hence $3^{\frac{(n-1)(n-2)}{2}}$ can serve as a lower bound for $|\succ_{\text{ir}}(n)|$. Since $3^{\frac{(n-1)(n-2)}{2}} = e^{\frac{(n-1)(n-2)}{2} \ln 3}$ and since

$$\lim_{n \rightarrow \infty} \left(\frac{(n-1)(n-2)}{2} \ln 3 - ((1 + \log_p n)n \ln r + kn) \right) = \infty,$$

we conclude that $e^{kn} \frac{|q_{\text{ir}}(n)|}{|\succ_{\text{ir}}(n)|} \rightarrow 0$ as $n \rightarrow \infty$. ■

The size of the domain n need not be at all large for the proportion of irrational preferences with quick checklists to be extremely small. To illustrate, the following table reports for $3 \leq n \leq 9$ the number of preference relations on X_n that fail to be complete and transitive, the number of those preferences that do not have quick 3-checklists, and the ratio of these two numbers, i.e., the proportion of irrational preferences on X_n with a quick 3-checklist. To keep the count under control, we give only the number of preferences – both for the total and for preferences with quick checklists – that are unique up to an order isomorphism.⁶

⁶Since there are only five relevant C_i orderings of C_i -indifference classes when $p = 3$, it is feasible to enumerate the irrational preferences on X_n that have quick 3-checklists and then weed out the preferences that are order isomorphic. I am indebted to Claire Blackman for writing code that performs this enumeration (it is available on request). The total number of preferences on X_n up to order isomorphism is given in Harary (1969); for the number of irrational preferences we subtract the number of rational preferences, which equals 2^{n-1} , from the Harary total.

n	# irrational preferences	# with quick 3-checklists	ratio
3	3	3	1
4	34	31	.91176471
5	566	226	.39929329
6	21,448	1,084	.05054084
7	2,142,224	3,847	.00179580
8	575,016,091	11,143	.00001938
9	415,939,242,776	217,902	.00000052

6 Minimum expected run time

While rational agents can always use checklists with the shortest possible length, the possibility remains that compared to their irrational peers rational agents may have to use checklists that in expectation proceed through more criteria before coming to a decision. Subject to one proviso this turns out not to be the case; rational agents not only can use the shortest checklists but can use checklists that on average execute the most quickly.

Suppose a preference \succ on a domain X with n elements has the checklist $C = (C_1, \dots, C_T)$. For any pair $\{x, y\} \subset X$, let $t_C(x, y) = t_C(y, x)$ be the index of the criterion that determines the agent's decision between x and y : $t_C(x, y)$ is the largest integer $i \in \{1, \dots, T\}$ such that (not $x C_j y$ and not $y C_j x$ for $j \in \{1, \dots, i-1\}$). If x and y are not ranked by \succ then $t_C(x, y) = T$. We define the *expected run time* of C to be

$$Et_C = \sum_{\{x,y\} \subset X: x \neq y} t_C(x,y) \frac{1}{2^{n(n-1)/2}}.$$

Since $2^{n(n-1)/2}$ is the number of unordered pairs in X , Et_C is the expectation of the number of criteria the agent must examine, assuming each pair is equally likely. We could apply any strictly increasing transformation to t without affecting the conclusion of the theorem below.

A preference \succ has *singleton indifference classes* if each \succ -indifference class contains a single element of X .

A preference \succ on X with singleton indifference classes has *minimum expected run time* if, for all $p \geq 2$, \succ has a p -checklist C such that, for any preference \succ' on X with singleton indifference classes and any p -checklist C' for \succ' , $Et_C \leq Et_{C'}$. Also, we say that \succ *has strictly smaller expected run time than* \succ' if, for all $p \geq 3$, \succ has a p -checklist C such that, for all p -checklists C' for \succ' , $Et_C \leq Et_{C'}$ and if strictly inequality holds for some $p \geq 3$, that is, if, for some $p \geq 3$, \succ has a p -checklist C such that, for all p -checklists C' for \succ' , $Et_C < Et_{C'}$.

Theorem 5 *Any rational preference \succ on X with singleton indifference classes has minimum expected run time. If a \succ' with singleton indifference classes on X lacks a quick p -checklist then \succ has strictly smaller expected run time than \succ' .*

By gluing Theorem 5 to Theorem 2 we conclude that for any irrational \succ there will be a subdomain on which \succ must have strictly smaller expected run time than a rational preference with singleton indifference classes restricted to the same subdomain.

A \succ with minimum expected run time in fact terminates very quickly: one may easily show that if \succ has minimum expected run time and C achieves this minimum for some $p \geq 2$ then $Et_C < 2$.

Why the restriction to singleton indifference classes? If C is a checklist for \succ and \succ has multiple-element indifference classes then any pair of indifferent elements will be decided only by the final criterion in C . So preferences with more pairs of indifferent options will run more slowly – even when they are rational. And even looking only at preferences with the same number of indifferent pairs, it could be that the quickest checklist for a preference decides between single-element indifference classes with its initial criteria and between multiple-element indifference classes with its later criteria, which again will slow average run time.

7 Conclusion

Rationality may not be as difficult as Herbert Simon supposed. If we identify a preference relation with a decision procedure and not with an agent who must uncover from his

mind a preexisting utility function, then there need be no intrinsic challenge to choosing rationally. For agents who use checklists, rational preferences will emerge if the constituent building blocks of a checklist – the criteria – are themselves rational; moreover criteria are necessarily rational when agents use simple binary criteria. Irrationality is still possible, of course, but there is no computational argument in its favor in the present setting. Indeed, it is the rational agents who hold the computational advantage. The agents who always use rational criteria in their checklists can make choice discriminations on some subdomains with strictly greater efficiency than the agents who use irrational criteria and they will never be less efficient on any subdomain.

A Appendix: Remaining proofs

Remainder of Proof of Theorem 2.

For the ‘only if’ direction of Theorem 2, suppose to the contrary that \succ fails to be complete and transitive but has uniformly quick checklists. In Part I, we assume that \succ has a subrelation \succ' on a domain X' such that X' has 10 \succ' -indifference classes and such that \succ' is complete but not transitive, while in Part II we consider the remaining cases. In both parts, we proceed by finding a subrelation \succ'' of \succ with domain $X'' \subset X'$ that has 9 \succ'' -indifference classes and that fails to have a quick 3-checklist, i.e., a checklist of length 2. The following Lemmas are useful in both parts.

Lemma 1. If \succ has 9 indifference classes and (C_1, C_2) is a 3-checklist for \succ , then (1) for any pair of distinct \succ -indifference classes I and J there does not exist a C_1 -equivalence class K_1 , a C_2 -equivalence class K_2 , a $x \in I$, and a $y \in J$, such that $\{x, y\} \subset (K_1 \cap K_2)$, (2) each C_i -equivalence class, $i = 1, 2$, equals the union of three \succ -indifference classes, and (3) if \succ is complete, then C_i is complete for $i = 1, 2$.⁷

Proof. The proof of Theorem 1 gives conclusion (1). Given (1), there can be no C_1 -equivalence class that contains a four element subset Y such that any $x, y \in Y$ are in distinct \succ -indifference classes. Since the C_1 -equivalence classes form a partition of X , there cannot be a C_1 -equivalence class that intersects only two or fewer \succ -indifference classes; if there were, there would then be a C_1 -equivalence class that intersects four or more \succ -indifference classes. Hence each C_1 -equivalence class equals the union of three \succ -indifference classes. If \succ is complete and there were two C_1 -equivalence classes I_1 and I'_1 such that neither $I_1 C_1 I'_1$ nor $I'_1 C_1 I_1$ holds, then each pair of the six \succ -indifference class in $I_1 \cup I'_1$ would have to be in distinct C_2 -equivalence classes, contradicting the fact that (C_1, C_2) is a 3-checklist.

Moving to C_2 , (1) implies that any two of the three \succ -indifference classes in a C_1 -equivalence class must be in distinct C_2 -equivalence classes. Since this conclusion applies

⁷Following Definition 5, the completeness of C_i means that, for every $x, y \in X$, either $x (C_i \cup \approx) y$ or $y (C_i \cup \approx) x$, where \approx is the equivalence relation of C_i .

to each of the three C_1 -equivalence classes and there are only three C_2 -equivalence classes, each C_2 -equivalence class must equal the union of three \succ -indifference classes, which completes the proof of (2). Since any two of the three \succ -indifference classes in a C_1 -equivalence class are in distinct C_2 -equivalence classes, for any pair of C_2 -equivalence classes I_2 and I'_2 there are two distinct \succ -indifference classes I and J in a single C_1 -equivalence class such that $I \subset I_2$ and $J \subset I'_2$. So if there were two C_2 -equivalence classes I_2 and I'_2 such that neither $I_2 C_2 I'_2$ nor $I'_2 C_2 I_2$ holds, there must be two distinct \succ -indifference classes I and J in a single C_1 -equivalence class such that neither $I C_2 J$ nor $J C_2 I$ holds; so \succ could not be complete. ■

Lemma 2. Suppose (i) (C_1, C_2) is a checklist for \succ , (ii) either $I C_1 J$ or $J C_1 I$ for any two distinct C_1 -equivalence classes I and J , (iii) K is a C_1 -equivalence class with $x, z \in K$, and (iv) $x \succ y \succ z$. Then $y \in K$.

Proof. If $y \notin K$ then y must be in a C_1 -equivalence class $L \neq K$ and $L C_1 J$ would contradict $y \succ z$ while $L C_1 K$ would contradict $x \succ y$. ■

Part I.

We assume that \succ has a subrelation \succ' on a domain X' with 10 \succ' -indifference classes such that \succ' is complete but not transitive. Without loss of generality, we suppose that $|X'| = 10$.

Since \succ' is complete, for any $X'' \subset X'$ each element of X'' is both a \succ' - and a \succ'' -indifference class, where \succ'' is the subrelation of \succ that has domain X'' . Fix a $X'' \subset X'$ with 9 elements and hence 9 \succ'' -indifference classes. Since \succ has uniformly quick checklists, there is a (C_1'', C_2'') that is a 3-checklist for \succ'' . By Lemma 1, there are three C_1'' -equivalence classes, say I''_α, I''_β , and I''_γ . Since the three elements in any of these C_1'' -equivalence classes must be in distinct C_2'' -equivalence classes, for any $\{i, j\} \subset \{\alpha, \beta, \gamma\}$, (I''_i, \succ) and (I''_j, \succ) must be order-isomorphic. Also by Lemma 1, C_2'' is complete. Let ω denote the sole element of $X' \setminus X''$.

(1) Suppose that each (I''_i, \succ) is order-isomorphic to the cycle, that is, to $(\{a, b, c\}, \{(a, b), (b, c), (c, a)\})$. For any I''_i , say I''_γ for concreteness, it is easy to confirm that there is a two-element subset $\{x, y\} \subset I''_\gamma$ such that $(\{x, y, \omega\}, \succ)$ is order-isomorphic to the line $(\{a, b, c\}, \{(a, b), (b, c), (a, c)\})$. Set $\widehat{X}'' = I''_\alpha \cup I''_\beta \cup \{x, y, \omega\}$.

By assumption, there is 3-checklist $(\widehat{C}_1'', \widehat{C}_2'')$ for the subrelation of \succ that has domain \widehat{X}'' . We show that I''_α and I''_β are \widehat{C}_1'' -equivalence classes. Case I. Some \widehat{C}_1'' -equivalence class \widehat{I}'' contains at least two elements of I''_i for either $i = \alpha$ or $i = \beta$. Lemma 2 then implies that \widehat{I}'' contains the third element of I''_i . Some other \widehat{C}_1'' -equivalence class \widehat{J}'' therefore contains at least two element of I''_j , $j = \{\alpha, \beta\} \setminus \{i\}$. Hence both I''_α and I''_β are \widehat{C}_1'' -equivalence classes. Case II. Each \widehat{C}_1'' -equivalence class contains at most one element of I''_i for both $i = \alpha$ and $i = \beta$. It follows that each of the three \widehat{C}_1'' -equivalence classes must contain exactly one element of I''_i , $i = \alpha, \beta$. Since (C_1'', C_2'') is a checklist for \succ'' , either $I''_\alpha \succ I''_\beta$ or $I''_\beta \succ I''_\alpha$. Choose indices so that $I''_\alpha \succ I''_\beta$. Let \widehat{I}'' be one of the \widehat{C}_1'' -equivalence classes and let $x, y \in \widehat{I}''$ be such that $x \in I''_\alpha$, $y \in I''_\beta$ and hence $x \succ y$. Since (I''_α, \succ) is order-isomorphic to the cycle there is a $z \in I''_\alpha$ such that $x \succ z$. Since $I''_\alpha \succ I''_\beta$, we have $z \succ y$, and so Lemma 2 implies that $z \in \widehat{I}''$, returning us to case I.

The three \widehat{C}_1'' -equivalence classes are therefore I''_α , I''_β , and $\{x, y, \omega\}$. But since

$(\{x, y, \omega\}, \succ)$ is not order-isomorphic to (I''_α, \succ) (or (I''_β, \succ)), we rule out the possibility that each (I''_i, \succ) is order-isomorphic to the cycle.

(2) Since \succ on I''_i is complete, the only other possibility is that each (I''_i, \succ) is order-isomorphic to the line $(\{a, b, c\}, \{(a, b), (b, c), (a, c)\})$. Since \succ'' is not transitive, C''_1 cannot be complete and transitive. Since, by Lemma 1, C''_1 is complete, the C''_1 ordering of C''_1 -equivalence classes must be a cycle.

Let $I''_\alpha, I''_\beta,$ and I''_γ continue to denote the C''_1 -equivalence classes; assign indices so that $I''_\alpha \succ I''_\beta \succ I''_\gamma \succ I''_\alpha$. We define three domains $\widehat{X}''_k, k = \alpha, \beta, \gamma,$ by omitting an arbitrary element of $x_k \in I''_k$ from X'' and replacing it ω . Let \widehat{I}''_k denote $(I''_k \cup \{\omega\}) \setminus \{x_k\}$.

Fix $k \in \{\alpha, \beta, \gamma\}$. For the 3-checklist $(\widehat{C}''_1, \widehat{C}''_2)$ that we assume exists for the subrelation of \succ that has domain \widehat{X}''_k , we show that for each \widehat{C}''_1 -equivalence class \widehat{I}'' there is a $I'' \in \{I''_\alpha, I''_\beta, I''_\gamma\}$ such that $\widehat{I}'' \setminus \{\omega\} \subset I''$. So the three \widehat{C}''_1 -equivalence classes must then be $\widehat{I}''_k, I''_i,$ and I''_j , where $\{i, j\} = \{\alpha, \beta, \gamma\} \setminus \{k\}$. To accomplish this, we exclude the possibilities (a) and (b) below. Note first that since the C''_1 ordering of C''_1 -equivalence classes is a cycle, the subrelation of \succ that has domain \widehat{X}''_k is not transitive.

(2a) Each \widehat{C}''_1 -equivalence class consists of a \succ -cycle. For either assignment of indices i and j so that $\{i, j\} = \{\alpha, \beta, \gamma\} \setminus \{k\}$ and for any \widehat{C}''_1 -equivalence class \widehat{I}'' , \widehat{I}'' obviously cannot equal either I''_i or I''_j since I''_i and I''_j are \succ -lines rather than \succ -cycles. Furthermore \widehat{I}'' cannot contain two elements of I''_i and one element of I''_j since then \widehat{I}'' would again be a \succ -line; hence each \widehat{C}''_1 -equivalence class contains one element of I''_i and one element of I''_j . But then for any pair of \widehat{C}''_1 -equivalence classes, \widehat{I}'' and \widehat{J}'' , we cannot have either $\widehat{I}'' \succ \widehat{J}''$ or $\widehat{J}'' \succ \widehat{I}''$ (as we must since \widehat{C}''_1 is complete and $(\widehat{C}''_1, \widehat{C}''_2)$ is a checklist for \succ on the domain \widehat{X}''_k).

(2b) Each \widehat{C}''_1 -equivalence class consists of a \succ -line but there is a \widehat{C}''_1 -equivalence class \widehat{I}'' such that there is no $I'' \in \{I''_\alpha, I''_\beta, I''_\gamma\}$ with $\widehat{I}'' \setminus \{\omega\} \subset I''$. Assigning the indices i and j so that $\{i, j\} = \{\alpha, \beta, \gamma\} \setminus \{k\}$, \succ is a linear order on $I''_i \cup I''_j$. So if, for each \widehat{C}''_1 -equivalence class \widehat{I}'' , $\widehat{I}'' \cap (I''_i \cup I''_j) \neq \emptyset$ then the \succ ordering of \widehat{C}''_1 -equivalence classes must be linear, contradicting the fact that the subrelation of \succ that has domain \widehat{X}''_k is not transitive. Hence only two \widehat{C}''_1 -equivalence classes intersect $I''_i \cup I''_j$. Since \succ is linear on $I''_i \cup I''_j$ and either $I''_i \succ I''_j$ or $I''_j \succ I''_i$, these two \widehat{C}''_1 -equivalence classes equal I''_i and I''_j .

Since for each \widehat{C}''_1 -equivalence class \widehat{I}'' there is a $I'' \in \{I''_\alpha, I''_\beta, I''_\gamma\}$ such that $\widehat{I}'' \setminus \{\omega\} \subset I''$, we can label the \widehat{C}''_1 -equivalence classes $\widehat{I}''_\alpha(k), \widehat{I}''_\beta(k),$ and $\widehat{I}''_\gamma(k)$, where, for $i \in \{\alpha, \beta, \gamma\}$, $\widehat{I}''_i(k) \setminus \{\omega\} \subset I''_i$. Since $I''_\alpha \succ I''_\beta \succ I''_\gamma, \widehat{I}''_\alpha(k) \succ \widehat{I}''_\beta(k) \succ \widehat{I}''_\gamma(k)$ for each $k \in \{\alpha, \beta, \gamma\}$. So, setting $k = \alpha, \omega \succ \widehat{I}''_\beta(\alpha) = I''_\beta$ and, setting $k = \gamma, I''_\beta = \widehat{I}''_\beta(\gamma) \succ \omega$, a contradiction.

Part II.

We turn to the remaining possibility where there is no $X' \subset X$ such that \succ' , the restriction of \succ to X' , defines 10 \succ' -indifference classes and is complete. We first show that in this case there is at least a $X' \subset X$ such that \succ' defines 10 \succ' -indifference classes.

Lemma 3. Let \succ be an asymmetric relation on a domain Y with $n \geq 3$ indifference classes and, for $Y' \subset Y$, let $\succ_{Y'}$ denote the restriction of \succ to Y' . If \succ_Y is incomplete then for any m with $3 \leq m \leq n$ there is a $Y' \subset Y$ such that (1) $|Y'| = m$, (2) Y' has m

$\succ_{Y'}$ -indifference classes, and (3) $\succ_{Y'}$ is incomplete.

Proof. Since the status of a $I \subset Y$ as an indifference class can vary according to the domain on which a subrelation of \succ is defined, in this proof we will always denote explicitly, by use of subscripts for \succ and for \sim , the relation that defines any indifference class. Note that for $Y'' \subset Y' \subset Y$ if x and y are in distinct $\succ_{Y''}$ -indifference classes then x and y are in distinct $\succ_{Y'}$ -indifference classes. So if Y'' has j $\succ_{Y''}$ -indifference classes then Y' has at least j $\succ_{Y'}$ -indifference classes.

Since \succ_Y has at least three \succ_Y -indifference classes and is incomplete, there exist $a, b \in Y$ such that $a \perp b$. (We use \perp to indicate that two alternatives are unranked by \succ : $x \perp y \Leftrightarrow (\text{not } x \succ y, \text{ not } y \succ x, \text{ and not } x \sim y)$. And, as in the proof of Theorem 1, \prec denotes $\{(x, y) \in X \times X : y \succ x\}$.) Since a and b are in separate \succ_Y -indifference classes, there exists some c such that, for at least one of the labelings ($s = a$ and $t = b$) and ($s = b$ and $t = a$), we have ($c \succ s$ and not $c \succ t$) or ($s \succ c$ and not $t \succ c$). So, defining $Y_3 = \{a, b, c\}$ and letting \succ_3 denote the restriction of \succ_Y to Y_3 , \succ_3 is incomplete and Y_3 has three \succ_3 -indifference classes. To argue by induction, suppose for any integer k between 3 and $m - 1$ that there is a $Y_k \subset Y$ with $|Y_k| = k$ such that, \succ_k , the restriction of \succ to Y_k , is incomplete and such that Y_k has k \succ_k -indifference classes. We show that there is a $Y_{k+1} \subset Y$ with $|Y_{k+1}| = k + 1$ such that \succ_{k+1} is incomplete and such that Y_{k+1} has $k + 1$ \succ_{k+1} -indifference classes. Let $x \in Y \setminus Y_k$ be such that, for all $w \in Y_k$, it is not the case that $x \sim_Y w$. If $Y_k \cup \{x\}$ cannot serve as the desired Y_{k+1} , then for the restriction of \succ_Y to $Y_k \cup \{x\}$, say $\succ_{k,x}$, we must have $x \sim_{k,x} y$ for some $y \in Y_k$. Given the induction premise that Y_k has k \succ_k -indifference classes, there can be at most one such y . Then, however, since not $x \sim_Y w$ for all $w \in Y_k$, there must be some $z \in Y \setminus (Y_k \cup \{x\})$ such that

$$(z \succ s \text{ and not } z \succ t) \text{ or } (s \succ z \text{ and not } t \succ z). \quad (\text{i})$$

for at least one of the labelings ($s = x$ and $t = y$), ($s = y$ and $t = x$). Since x and y are in distinct $\succ_{Y_k \cup \{x, z\}}$ -indifference classes, and each pair in Y_k are in distinct \succ_{Y_k} -indifference classes, $Y_k \cup \{x, z\}$ must have either $k + 1$ or $k + 2$ $\succ_{Y_k \cup \{x, z\}}$ -indifference classes (see the first paragraph of the proof). In the former case, $Y_k \cup \{x, z\}$ can serve as Y_{k+1} . So suppose $Y_k \cup \{x, z\}$ has $k + 2$ $\succ_{Y_k \cup \{x, z\}}$ -indifference classes. We show that, for either $Z = Y_k \cup \{z\}$ or $Z = (Y_k \cup \{x, z\}) \setminus \{y\}$, Z has $k + 1$ \succ_Z -indifference classes. Given that for either selection of Z , we have either $\{a, b, c\} \subset Z$ or, if $y \in \{a, b, c\}$, $\{a, b, c, x\} \setminus \{y\} \subset Z$, the Lemma is then proved. Since both Y_k and $(Y_k \cup \{x\}) \setminus \{y\}$ have k indifference classes, Z can fail to have $k + 1$ \succ_Z -indifference classes only if Z has k \succ_Z -indifference classes and $z \sim_Z v$ for some $v \in Z \setminus \{z\}$. Suppose $Y_k \cup \{z\}$ in fact has k $\succ_{Y_k \cup \{z\}}$ -indifference classes. Then for the $v \neq z$ such that $z \sim_{Y_k \cup \{z\}} v$ either $v \in Y_k \setminus \{y\}$ or $v = y$. If the first case, $v \in Y_k \setminus \{y\}$, obtains then $zRy \Leftrightarrow vRy$ for both $R = \succ$ and $R = \prec$ and so, by (i) and the fact $x \sim_{Y_k \cup \{x\}} y$, it is not the case that $zRx \Leftrightarrow vRx$ for both $R = \succ$ and $R = \prec$; hence not $z \sim_{(Y_k \cup \{x, z\}) \setminus \{y\}} v$. Furthermore we cannot have either $z \sim_{(Y_k \cup \{x, z\}) \setminus \{y\}} x$ or $z \sim_{(Y_k \cup \{x, z\}) \setminus \{y\}} v'$ for $v' \in Y_k \setminus \{v\}$ since then, given that $x \sim_{Y_k \cup \{x\}} y$, we would have, respectively, $y \sim_{Y_k} v$ or $v' \sim_{Y_k} v$, contradicting the induction premise that Y_k has k \succ_{Y_k} -indifference classes. Thus $(Y_k \cup \{x, z\}) \setminus \{y\}$ has $k + 1$ $\succ_{(Y_k \cup \{x, z\}) \setminus \{y\}}$ -indifference classes and may serve as Y_{k+1} . The second case, $v = y$, is similar. Then (i) implies not $z \sim_{(Y_k \cup \{x, z\}) \setminus \{y\}} x$. It furthermore cannot be the case that $z \sim_{(Y_k \cup \{x, z\}) \setminus \{y\}} v'$ for $v' \in Y_k \setminus \{y\}$ since then $y \sim_{Y_k} v'$, again violating the assumption

that Y_k has $k \succ_{Y_k}$ -indifference classes. So again $(Y_k \cup \{x, z\}) \setminus \{y\}$ may serve as Y_{k+1} . ■

For subrelations \succ_* of \succ on $X_* \subset X$, we now drop the cumbersome clarification needed in the proof of Lemma 3 between \succ_* -indifference classes and \succ -indifference classes. We have no further need to consider indifference classes defined from \succ on the entire domain X ; so instead we will always name the subdomain $X_* \subset X$ explicitly and refer to a \succ_* -indifference class simply as a \succ -indifference class.

Lemma 3 allows us to assume that \succ on X' has 10 \succ -indifference classes and is incomplete. We again suppose, without loss of generality, that $|X'| = 10$. By setting $Y = X'$, Lemma 3 also implies that there is a nine-element $X'' \subset X'$ such that \succ on X'' has nine \succ -indifference classes and such that \succ on X'' is incomplete. Since the proof is done if \succ on X'' fails to have a 3-checklist of length 2, suppose it does have such a checklist. Since \succ on X'' is incomplete, it follows from Lemma 1 that for either $i = 1$ or 2 (or both) C_i must be incomplete. Since Lemma 1 also implies that each C_i defines three C_i -equivalence classes, the C_i ordering of C_i -equivalence classes must be order-isomorphic to one of two incomplete orderings: letting $\{a, b, c\}$ denote the C_i -equivalence classes, these two orderings are $\{(a, b)\}$, which we label Inc(1), and $\{(a, b), (b, c)\}$, which we label Inc(2). Let Com denote either one of the two complete orderings of $\{a, b, c\}$ (the line and cycle). A 3-checklist of length 2 must therefore take the following form: either C_1 - C_2 or C_2 - C_1 must be of type Inc(1)-Inc(1), Inc(2)-Inc(2), Inc(1)-Inc(2), Inc(1)-Com, or Inc(2)-Com, where in each case ' C_i - C_j is of type A-B' means that the C_i (resp. C_j) ordering of C_i (resp. C_j)-equivalence classes is order-isomorphic to A (resp. B). Since the cases are broadly similar, we consider only one case – where C_1 - C_2 is of type Inc(2)-Com – in fine detail.

Before proceeding to Inc(2)-Com, we list, for each of the possibilities given in the previous paragraph, the number of elements in X'' that are ranked relative to each $x \in X''$, which in the terminology of graph theory is called the *degree of x* . Given a set Y with 9 elements and a 3-checklist (C', C'') that is quick for some asymmetric relation $\succ_{(C', C'')}$ that has domain Y and that has nine indifference classes, we define the degree of x , $d(x)$, to be $|\{y \in Y : y \succ_{(C', C'')} x \text{ or } x \succ_{(C', C'')} y\}|$ for each of the nine $x \in Y$. When the nine possible $d(x)$, one for each $x \in Y$, are arranged in decreasing order, they are called a *degree sequence*. If we instead began with the 3-checklist (C'', C') that is quick for some $\succ_{(C'', C')}$ with the same domain Y and also with nine indifference classes, we would arrive at the same degree sequence. There are therefore five possible degree sequences, each simple to calculate, which we list in the following table:

Table of degree sequences	
Inc(1)-Inc(1)	5,5,5,5,3,3,3,3,0
Inc(2)-Inc(2)	8,7,7,7,7,5,5,5,5
Inc(1)-Inc(2) or Inc(2)-Inc(1)	7,7,6,5,5,5,5,3,3
Inc(1)-Com or Com-Inc(1)	7,7,7,7,7,7,6,6,6
Inc(2)-Com or Com-Inc(2)	8,8,8,7,7,7,7,7,7

Inc(2)-Com. We assume there is a $X'' \subset X'$ with $|X''| = 9$ that has a 3-checklist (C_1, C_2) of length 2 where the C_1 ordering of C_1 -equivalence classes is of type Inc(2) and the C_2 ordering of C_2 -equivalence classes is of type Com. Applying Lemma 1, without

loss of generality we can assign labels $a, b, c, d, e, f, g, h,$ and i to the elements of X'' so that $\{a, b, c\}C_1\{d, e, f\}C_1\{g, h, i\}$ and the three C_2 -equivalence classes are $\{a, d, g\}, \{b, e, h\},$ and $\{c, f, i\}$. For later use, we note the \succ -unranked pairs for the \succ for which (C_1, C_2) is a checklist: $a \perp g, b \perp h,$ and $c \perp i$. Consider the domain of 9 elements $\widehat{X}''(z)$ formed by replacing one element $z \in \{a, b, c, d, e, f, g, h, i\}$ with the tenth element of X' which we label ω . We first show that if $\widehat{X}''(z)$ has a 3-checklist of length 2 then it is either of type Inc(2)-Com or Com-Inc(2). Suppose to begin that $\widehat{X}''(z)$ has nine \succ -indifference classes. Since the degree of any element in both X'' and $\widehat{X}''(z)$ can differ by at most one, the presence in the degree sequence table of numbers 5 or smaller for the checklists Inc(1)-Inc(1), Inc(2)-Inc(2), Inc(1)-Inc(2), and Inc(2)-Inc(1) rules out these possibilities as quick checklists when the domain is $\widehat{X}''(z)$. The remaining possibilities besides Inc(2)-Com and Com-Inc(2) are Inc(1)-Com and Com-Inc(1). But in order for $\widehat{X}''(z)$ to have a Inc(1)-Com or Com-Inc(1) checklist, it must be that ω is not ranked against $d, e,$ or f if they are in $\widehat{X}''(z)$ (these are the elements of X'' with degree eight) and is also not ranked against three elements of $a, b, c, g, h,$ and i if they are in $\widehat{X}''(z)$ (these are the elements of X'' with degree seven). Given that one element of X'' is not an element of $\widehat{X}''(z)$, we conclude that ω could have a degree of at most four in $\widehat{X}''(z)$, which is not consistent with the Inc(1)-Com degree sequence. Now let $\widehat{X}''(z)$ have fewer than nine \succ -indifference classes. It is easy to confirm that $\{a, b, c, d, e, f, g, h, i\} \setminus \{z\}$ must have eight \succ -indifference classes; so the only case to consider is where $\widehat{X}''(z)$ has eight \succ -indifference classes. If a checklist is Inc(1)-Com or Com-Inc(1) and has eight indifference classes then one can easily confirm that the degree sequence is either $(6, 6, 6, 6, 6, 6, 5, 5)$ or $(7, 7, 6, 6, 6, 6, 5, 5)$ while if a checklist is Inc(2)-Com or Com-Inc(2) then the degree sequence is $(7, 7, 6, 6, 6, 6, 6, 6)$ or $(7, 7, 7, 7, 6, 6, 6, 6)$. Therefore it could not be that a $\widehat{X}''(z)$ with eight indifference classes has a Inc(1)-Com or Com-Inc(1) checklist.

To exclude Com-Inc(2), observe that if there exists a Com-Inc(2) checklist $(\widehat{C}_1, \widehat{C}_2)$ when the domain is $\widehat{X}''(z)$ then a and g (resp. b and h) if they are elements of $\widehat{X}''(z)$ must be in the same \widehat{C}_1 -equivalence class since $a \perp g$ (resp. $b \perp h$) and \widehat{C}_1 is complete. Now $a \succ \{d, e, f\} \succ g$ and $b \succ \{d, e, f\} \succ h$. Since either $\{a, g\} \subset \widehat{X}''(z)$ or $\{b, h\} \subset \widehat{X}''(z)$ (or both) and \widehat{C}_1 is complete, Lemma 2 implies that if $\{d, e, f\} \subset \widehat{X}''(z)$ then $d, e,$ and f must be in the same \widehat{C}_1 -equivalence class as $\{a, g\}$ or $\{b, h\}$ (whichever pair is a subset of \widehat{X}''). Since at least two elements of $\{d, e, f\}$ must be in $\widehat{X}''(z)$, we have a \widehat{C}_1 -equivalence class with at least four elements, in violation of Lemma 1.

Having established that if $\widehat{X}''(z)$ has a 3-checklist of length 2 the checklist is of type Inc(2)-Com, we proceed to identify a $\widehat{X}''(z)$ that does not have a Inc(2)-Com checklist. Suppose $(\widehat{C}_1, \widehat{C}_2)$ is a 3-checklist for $\widehat{X}''(z)$. Let the three \widehat{C}_1 -equivalence classes be labeled $I, J,$ and K where $I \widehat{C}_1 J \widehat{C}_1 K$. Since \widehat{C}_2 is complete and since $a \perp g, b \perp h,$ and $c \perp i$, any of the pairs $(a, g), (b, h),$ and (c, i) that is a subset of $\widehat{X}''(z)$ must be (i) contained in a single \widehat{C}_2 -equivalence class and in a single C_2 -equivalence class, and (ii) one element of the pair must be I and the other in K . Suppose $z \in \{a, b, c, g, h, i\}$, let y be the unique $y \in \{a, b, c, g, h, i\}$ such that $y \perp z$, and let (s, t) be one of the pairs in $\{(a, g), (b, h), (c, i)\}$ such that $(s, t) \subset \widehat{X}''(z)$. Since (i) $\{s, t\}$ is a subset of a C_2 -equivalence class, (ii) $\{s, t, y\} \cap \{d, e, f\} = \emptyset$, and (iii) C_2 is complete, we conclude that either $(y \succ s$ and $y \succ t)$

or $(s \succ y$ and $t \succ y)$. Since $s \perp t$, we must have $\{s, t\} \subset I \cup K$, and we conclude that $y \notin J$. It follows that $J = \{d, e, f\}$ since otherwise $u \perp v$ for some $u \in \{d, e, f\}$ and some $v \in \{a, b, c, g, h, i\} \setminus \{z\}$. Hence, since $\{a, b, c\} \succ \{d, e, f\} \succ \{g, h, i\}$, $\{a, b, c\} \setminus \{z\} \subset I$ and $\{g, h, i\} \setminus \{z\} \subset K$.

Therefore if $z = c$, $\widehat{X}''(z)$ consists of nine \succ -indifference classes we conclude, since $J = \{d, e, f\}$ and $K = \{g, h, i\}$, that $\omega \in I$ and so $\omega \succ \{d, e, f\}$. If for $z = i$, $\widehat{X}''(z)$ also consists of nine \succ -indifference classes then $\omega \in K$ and therefore $\{d, e, f\} \succ \omega$, a contradiction.

It remains to dispose of the possibility that for either $z = c$ or $z = i$, $\widehat{X}''(z)$ consists of less than nine \succ -indifference classes. We use the following easily confirmed fact: for any domain Y that contains at least 7 elements of X'' , any 2 elements of $X'' \cap Y$ are members of distinct \succ -indifference classes. So we need only consider the case where, for either $z = c$ or $z = i$, $\widehat{X}''(z)$ consists of eight \succ -indifference classes. There must then be a $x \in X'' \setminus \{z\}$ such that (i) if zRx , where $R \in \{\succ, \prec, \perp\}$, then not $zR\omega$ and (ii) for any $w \in (X'' \cup \{\omega\}) \setminus \{z\}$ and $R \in \{\succ, \prec\}$, $wR\omega \Leftrightarrow wRx$. We next show that $\widehat{X}''(v)$ must consist of nine \succ -indifference classes for any $v \in X'' \setminus \{z\}$. To do so, fix a $v \in X'' \setminus \{z\}$ and suppose $\widehat{X}''(v)$ has fewer than nine \succ -indifference classes. There must then be a $u \in X'' \setminus \{v\}$ such that (i*) if vRu , where $R \in \{\succ, \prec, \perp\}$, then not $vR\omega$ and (ii**) for any $w \in (X'' \cup \{\omega\}) \setminus \{v\}$ and $R \in \{\succ, \prec\}$, $wR\omega \Leftrightarrow wRu$. Since $z \in \widehat{X}''(v)$, and given (i) and (ii*), $u \neq x$. But then for the domain $(X'' \setminus \{z, v\}) \cup \{\omega\}$, ω , x , and u are all elements of the same \succ -indifference class, and hence x and u are elements of the same \succ -indifference class for the same domain, contradicting the fact mentioned at the beginning of the paragraph. To finish the Inc(2)-Com case, we repeat the argument made in the previous paragraph: since, for $z = a$, $\widehat{X}''(z)$ consists of nine \succ -indifference classes, $\omega \in I$ and so $\omega \succ \{d, e, f\}$ and, since, for $z = c$, $\widehat{X}''(z)$ must also consist of nine \succ -indifference classes, $\omega \in K$ and therefore $\{d, e, f\} \succ \omega$, a contradiction.

For each of the remaining checklist types, one may use the degree sequence table to argue that for any $z \in X''$ if a quick checklist exists when the domain is $\widehat{X}''(z)$ then it must be of the same type as the checklist when the domain is X'' . Below, we provide the key step that there are at least two items $x, y \in X''$ such that the domains $\widehat{X}''(x)$ and $\widehat{X}''(y)$ cannot both have quick checklists, on the assumption that both domains contain 9 \succ -equivalence classes. If some $\widehat{X}''(z)$ has 8 \succ -equivalence classes, then as in the previous paragraph one may always find an alternative $z' \in X'' \setminus \{z\}$ such that $\widehat{X}''(z')$ has no quick checklist.

In each case, let $X' = \{a, b, c, d, e, f, g, h, i, \omega\}$, let $X'' = X' \setminus \{\omega\}$, and let the quick checklist that we assume exists for the domain X'' be denoted (C_1, C_2) . The criteria for the quick checklist for $\widehat{X}''(z)$ is denoted $(\widehat{C}_1, \widehat{C}_2)$.

Inc(2)-Inc(1). Choose labels so that $\{a, b, c\}C_1\{d, e, f\}C_1\{g, h, i\}$ and $\{a, d, g\}C_2\{b, e, h\}$. Let I, J, K denote the three \widehat{C}_1 -equivalence classes, where $I\widehat{C}_1J\widehat{C}_1K$. Consider $\widehat{X}''(z)$ for $z = c$ and $z = i$. Then d and e , which have the largest degree when the domain is X'' , will continue to have the largest degree and must therefore be elements of J . The specification of (C_1, C_2) implies $\{a, b, c\} \succ \{d, e, f\} \succ \{g, h, i\}$, $a \succ b$, $d \succ e$, and $g \succ h$. When the domain is $\widehat{X}''(z)$, $\{a, b, c\} \setminus \{z\} \subset I$ since otherwise, given that \widehat{C}_2 is Inc(1), the element of $\{a, b, c\} \setminus \{z\}$ not in I could not be \succ -superior to both d and e for the \succ that

has the checklist $(\widehat{C}_1, \widehat{C}_2)$. Similarly $\{g, h, i\} \setminus \{z\} \subset K$. And since $\{a, b\} \succ \{f\}$, we have $f \in J$ since otherwise (again using the fact that \widehat{C}_2 is Inc(1)) f could not be \succ -inferior to both a and b for the \succ that has the checklist $(\widehat{C}_1, \widehat{C}_2)$. So, for $z = c$, we have $\omega \in I$ and therefore $\omega \succ \{d, e, f\}$ and, for $z = i$, we have $\omega \in K$ and therefore $\{d, e, f\} \succ \omega$, a contradiction.

Inc(2)-Inc(2). Choose labels so that $\{a, b, c\}C_1\{d, e, f\}C_1\{g, h, i\}$, and let I, J, K denote the three \widehat{C}_1 -equivalence classes, where $I\widehat{C}_1J\widehat{C}_1K$. For $z = c$ and $z = i$, the domain $\widehat{X}''(z)$ has $J = \{d, e, f\}$. For $z = c$, we have $\omega \in I$ and therefore $\omega \succ \{d, e, f\}$ and, for $z = i$, we have $\omega \in K$ and therefore $\{d, e, f\} \succ \omega$.

Com-Inc(2) and Com-Inc(1). The conclusions for Inc(2)-Inc(2) continue to hold using the same labeling.

Inc(1)-Com and Inc(1)-Inc(2). Choose labels so that $\{a, b, c\}C_1\{d, e, f\}$, $\{a, d, g\}C_2\{b, e, h\}C_2\{c, f, i\}$, and let I, J, K denote the three \widehat{C}_2 -equivalence classes, where $I\widehat{C}_2J\widehat{C}_2K$. For $z = g$ and $z = i$, the domain $\widehat{X}''(z)$ has $J = \{b, e, h\}$. For $z = g$, we have $\omega \in I$ and therefore $\omega \succ \{b, e, h\}$ and, for $z = i$, we have $\omega \in K$ and therefore $\{b, e, h\} \succ \omega$.

Inc(1)-Inc(1). The easiest argument for the final case is slightly different. Choose labels so that $\{a, b, c\}C_1\{d, e, f\}$ and $\{a, d, g\}C_2\{b, e, h\}$. For $\widehat{X}''(i)$, if ω has $\omega \succ x$ or $x \succ \omega$ for any $x \in \{a, b, c, d, e, f, g, h\}$ then no element of $\widehat{X}''(i)$ is \succ -unranked relative to all other elements of $\widehat{X}''(i)$, in violation of the list $(5, 5, 5, 5, 3, 3, 3, 3, 0)$ given by the degree sequence table. If $\omega \perp x$ for all $x \in \{a, b, c, d, e, f, g, h\}$ then $\widehat{X}''(x)$ for any $x \in \{a, b, c, d, e, f, g, h\}$ has a degree sequence that differs from $(5, 5, 5, 5, 3, 3, 3, 3, 0)$. ■

Proof of Theorem 5.

Let \succ on the domain X be a rational preference with n singleton indifference classes and fix some $p \geq 2$. We recursively define a p -checklist $\widehat{C} = (\widehat{C}_1, \dots, \widehat{C}_{\lceil \log_p n \rceil})$ for \succ as follows.

Set $\widehat{C}_0 = \emptyset$. Given the binary relation \widehat{C}_i on X , let $\widehat{\mathcal{I}}_i$ denote the partition of X whose cells are the \widehat{C}_i -equivalence classes. So in particular $\widehat{\mathcal{I}}_0 = \{X\}$. Given the partitions $\widehat{\mathcal{I}}_0, \dots, \widehat{\mathcal{I}}_r$ of X , we let $\widehat{\mathcal{J}}_r = \bigvee_{j=0, \dots, r} \widehat{\mathcal{I}}_j$ denote the coarsest common refinement of $\widehat{\mathcal{I}}_0, \dots, \widehat{\mathcal{I}}_r$. For the recursion, take $\widehat{C}_0, \dots, \widehat{C}_{i-1}$ as given and define for each $J \in \widehat{\mathcal{J}}_{i-1}$ a partition $\{J(1), \dots, J(q)\}$ of J , where $q = \min[p, |J|]$, such that (1) each cell contains either $\lfloor \frac{|J|}{q} \rfloor$ or $\lceil \frac{|J|}{q} \rceil$ elements (where $\lfloor a \rfloor$ is the largest integer $\leq a$) and (2) $J(r) \succ J(s)$ if $r < s$. If $|J| < p$, set $J(m) = \emptyset$ for $|J| < m \leq p$. For any $k = 1, \dots, p$ such that $\bigcup_{J \in \widehat{\mathcal{J}}_{i-1}} J(k) \neq \emptyset$, define a \widehat{C}_i -equivalence class by $\widehat{I}_i(k) = \bigcup_{J \in \widehat{\mathcal{J}}_{i-1}} J(k)$ and then set \widehat{C}_i by $I_i(r) \widehat{C}_i I_i(s)$ if and only if $r < s$ and both $\widehat{I}_i(r)$ and $\widehat{I}_i(s)$ are nonempty. It is easy to check that each $J \in \widehat{\mathcal{J}}_{\lceil \log_p n \rceil}$ is a singleton and that $\widehat{C} = (\widehat{C}_1, \dots, \widehat{C}_{\lceil \log_p n \rceil})$ is a p -checklist for \succ .

Using the fact that $\lfloor \frac{\lfloor n/p \rfloor}{p} \rfloor = \lfloor \frac{n}{p^2} \rfloor$ and $\lceil \frac{\lceil n/p \rceil}{p} \rceil = \lceil \frac{n}{p^2} \rceil$, it follows that for any $i = 1, \dots, \lceil \log_p n \rceil$ and any $J, K \in \widehat{\mathcal{J}}_i$, $\| |J| - |K| \| \leq 1$. It will be useful for the final paragraph to note that if a partition \mathcal{L} of X contains only cells with cardinalities that differ by at most 1 then \mathcal{L} contains only sets with either $\lfloor \frac{n}{|\mathcal{L}|} \rfloor$ or $\lceil \frac{n}{|\mathcal{L}|} \rceil$ elements.

Let \succ' be an arbitrary preference on X and let $C' = (C'_1, \dots, C'_T)$ be a p -checklist for \succ' . It is sufficient to show that for any $\tau \leq T$ the first τ criteria in C' make no more

discriminations than do the first $\min[\tau, \lceil \log_p n \rceil]$ criteria in \widehat{C} . Let τ_{\min} henceforth denote $\min[\tau, \lceil \log_p n \rceil]$. That is, if we define, for an arbitrary checklist C ,

$$D_C(\tau) = \{\{x, y\} \subset X : \exists j \text{ with } 1 \leq j \leq \tau \text{ such that either } x C_j y \text{ or } y C_j x\},$$

it is sufficient to show that $|D_{\widehat{C}}(\tau_{\min})| \geq |D_{C'}(\tau)|$ for any $\tau \leq T$.

Before proceeding, define for any partition \mathcal{L} of X the set

$$D_{\mathcal{L}} = \{\{x, y\} \subset X : \exists I, J \in \mathcal{L} \text{ such that } x \in I, y \in J, \text{ and } I \neq J\}.$$

Given criteria (C_1, \dots, C_t) the equivalence classes of each criterion C_j define a partition \mathcal{I}_j of X , and therefore $\mathcal{J}_t = \bigvee_{j=1, \dots, t} \mathcal{I}_j$ and $D_{\mathcal{J}_t}$ are also well-defined. Since x and y are in different cells of \mathcal{J}_t only if $x C_j y$ for some $j \in \{1, \dots, t\}$, $|D_{\mathcal{J}_t}| \geq |D_C(t)|$. Since inequality can occur only if some C_j is incomplete, for any initial segment of the criteria in the \widehat{C} we defined for \succ , say $\widehat{C}_1, \dots, \widehat{C}_i$, we have $|D_{\widehat{\mathcal{J}}_i}| = |D_{\widehat{C}}(i)|$. Notice also that if each C_j in (C_1, \dots, C_t) has at most p equivalence classes then \mathcal{J}_t can have at most $\min[p^t, n]$ cells but $\widehat{\mathcal{J}}_i$ has exactly $\min[p^i, n]$ cells.

To get our result, fix some $\tau \leq T$. For $i = 1, \dots, \tau$, let \mathcal{I}'_i denote the partition formed by the equivalence classes of C'_i . Beginning with $\mathcal{J}'_{\tau} = \bigvee_{j=1, \dots, \tau} \mathcal{I}'_j$, we define a new partition by taking any pair $I, J \in \mathcal{J}'_{\tau}$ of maximal and minimal cardinality and replacing them with two sets that partition $I \cup J$ into sets with cardinalities $\lfloor \frac{|I \cup J|}{2} \rfloor$ and $\lceil \frac{|I \cup J|}{2} \rceil$. By iterating this operation, we arrive at a partition \mathcal{K} such that if $K, J \in \mathcal{K}$ then $||K| - |J|| \leq 1$ and where $|\mathcal{K}| = |\mathcal{J}'_{\tau}|$. It is easy to confirm that when this operation is applied to \mathcal{L}_1 to generate \mathcal{L}_2 , we have $|D_{\mathcal{L}_2}| \geq |D_{\mathcal{L}_1}|$. So $|D_{\mathcal{K}}| \geq |D_{\mathcal{J}'_{\tau}}|$. As noted earlier, each $K \in \mathcal{K}$ has $\lfloor \frac{n}{|\mathcal{K}|} \rfloor$ or $\lceil \frac{n}{|\mathcal{K}|} \rceil$ elements and each $J \in \widehat{\mathcal{J}}_{\tau_{\min}}$ has $\lfloor \frac{n}{|\widehat{\mathcal{J}}_{\tau_{\min}}|} \rfloor$ or $\lceil \frac{n}{|\widehat{\mathcal{J}}_{\tau_{\min}}|} \rceil$ elements. Hence, since $|\mathcal{K}| \leq \min[p^{\tau}, n]$ and $|\widehat{\mathcal{J}}_{\tau_{\min}}| = \min[p^{\tau}, n]$, we have $|K| \geq |J|$ for any $K \in \mathcal{K}$ and $J \in \widehat{\mathcal{J}}_{\tau_{\min}}$. It follows that $|D_{\widehat{C}}(\tau)| = |D_{\widehat{\mathcal{J}}_{\tau_{\min}}}| \geq |D_{\mathcal{K}}|$. Since $|D_{\mathcal{J}'_{\tau}}| \geq |D_{C'}(\tau)|$, we have $|D_{\widehat{C}}(\tau_{\min})| \geq |D_{C'}(\tau)|$ as desired.

As for the second sentence of the theorem, suppose \succ' on X lacks a quick p -checklist. Then $t_{C'}(x, y) = T$ for some $\{x, y\} \subset X$ but $t_C(x, y) < T$ for all $\{x, y\} \subset X$. Combined with the fact that $|D_{\widehat{C}}(\tau_{\min})| \geq |D_{C'}(\tau)|$ for any $\tau < T$, \succ must have strictly smaller expected run time than \succ' . ■

References

- [1] Abramowitz, M. and I. Stegun (eds.), 1972, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover: New York.
- [2] Apestequia J. and M. Ballester, 2005, ‘Minimal books of rationales’ available at <ftp://ftp.econ.unavarra.es/pub/DocumentosTrab/DT0501.PDF>.
- [3] Apestequia J. and M. Ballester, 2008a, ‘A characterization of sequential rationality’ available at SSRN: <http://ssrn.com/abstract=1143272>.

- [4] Apesteagua J. and M. Ballester, 2008b, ‘On the complexity of rationalizing behavior’ available at <http://www.econ.upf.edu/~apesteagua/Documentos/complexity.pdf>
- [5] Chipman, J., 1960, ‘The foundations of utility’ *Econometrica* 28: 193-224.
- [6] 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.
- [7] Cuesta Dutari, N., 1943, ‘Teoria decimal de los tipos de orden’ *Revista Matematica Hispano-Americana* 3: 186-205, 242-268.
- [8] Cuesta Dutari, N., 1947, ‘Notas sobre unos trabajos de Sierpinski’ *Revista Matematica Hispano-Americana* 7: 128-131.
- [9] Fishburn, P., 1970, *Utility Theory for Decision Making*. Wiley: New York.
- [10] Fishburn, P., 1974, ‘Lexicographic orders, utilities and decision rules: a survey’ *Management Science* 20: 1442-1471.
- [11] Goldberg, K., M. Newman, and E. Haynsworth, 1972, ‘Combinatorial analysis’ in M. Abramowitz and I. Stegun (eds.) *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover: New York.
- [12] Harary, F., 1969, *Graph Theory*. Addison-Wesley, Reading, MA.
- [13] Kalai, G., A. Rubinstein, and R. Spiegler, 2002, ‘Rationalizing choice functions by multiple rationales’ *Econometrica* 70: 2481-88.
- [14] Knuth, D., 1997, *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*. Addison-Wesley: Reading MA.
- [15] Mandler, M., P. Manzini, and M. Mariotti, 2008, ‘A million answers to twenty questions: choosing by checklist’ available at <http://personal.rhul.ac.uk/uhte/035/million.answers.to.twenty.questions.pdf>
- [16] Mandler, M., 2009, ‘Rationality and the speed of decision-making’ in A. Heifetz (ed.) *Proceedings of the 12th Conference on Theoretical Aspects of Rationality and Knowledge (TARK XII)*. ACM Digital Library. Available at <http://personal.rhul.ac.uk/uhte/035/Mandler.TARK.2009.pdf>
- [17] Manzini, P. and M. Mariotti, 2007, ‘Sequentially rationalizable choice’ *American Economic Review* 97: 1824-1839.
- [18] Salant, Y., 2003, ‘Limited computational resources favor rationality’ mimeo.
- [19] Simon, H., 1976, ‘From substantive to procedural rationality’ in S. Latsis (ed.) *Methods and Appraisal in Economics*. Cambridge University Press, Cambridge.
- [20] Simon, H., 1990, ‘Bounded rationality’ in J. Eatwell, M. Milgate and P. Newman (eds.) *The New Palgrave: Utility and Probability*. Macmillan, London.

- [21] Tversky, T., and I. Simonson, 'Context-dependent preferences' *Management Science* 39: 1179-1189.