



Munich Personal RePEc Archive

How mindless is standard economics really?

Schipper, Burkhard C
University of California, Davis

22. October 2009

Online at <http://mpa.ub.uni-muenchen.de/18080/>
MPRA Paper No. 18080, posted 23. October 2009 / 04:13

HOW MINDLESS IS STANDARD ECONOMICS REALLY?

Burkhard C. Schipper*

October 22, 2009

Abstract

Contrary to claims by Gul and Pesendorfer (2008), I show that standard economics makes use of non-choice evidence in a meaningful way. This is because standard economics solely grounded in the theory of choice is “incomplete”. That is, it has content that can not be revealed with any general choice procedure.

Keywords: Revealed preference, theory of choice, neuroeconomics, non-choice evidence, machines.

JEL-Classifications: A12, B41, C80, C90, D60, D80, D87.

*Department of Economics, University of California, Davis. Email: bcschipper@ucdavis.edu
I thank Giacomo Bonanno, Joseph Cummins, Klaus Nehring, Till Stegers and participants in the Workshop on Information Processing, Rational Belief Change and Social Interaction in Dagstuhl 2009 for helpful comments.

1 Introduction

In this note I discuss some claims in Gul and Pesendorfer's (2008) "The Case for Mindless Economics". I do not take an issue here with their treatment of neuroeconomics or behavioral economics, but rather with what they call "standard economics". I will show that by admitting only choice-evidence as Gul and Pesendorfer (2008) advocate, standard economics is "incomplete" in the sense that it has content that can not be tested with any general choice procedure. Thus, non-choice evidence may be meaningful for standard economics.

Gul and Pesendorfer's (2008) article facilitated a recent fruitful discourse on economic methodology. On one hand, there is neuroeconomics - actively collecting non-choice data - and behavioral economics - featuring notions that may be difficult to tie satisfactory to choice-evidence only. On the other hand, there is the revealed preference approach. A collection of articles on this debate can be found in the 2008 special issue of *Economics & Philosophy* on neuroeconomics or the volume edited by Caplin and Schotter (2008) in which Gul and Pesendorfer's article appeared.

2 A Simple Gedankenexperiment

Consider the following Gedankenexperiment: Suppose we are told that an "entity" is contained in a black box. We can not see what is inside the black box. All we observe is that the black box selected alternative x among the set of alternatives $\{x, y\}$.

In their discussion of "standard economics", Gul and Pesendorfer (2008, p. 8) write that "(a)lternative x is deemed to be better than alternative y if and only if, given the opportunity, the individual would choose x over y . Hence, welfare is defined to be synonymous with choice behavior." So, according to Gul and Pesendorfer (2008), we may conclude that the "entity" maximized its utility or welfare by selecting x . Indeed, they write (p. 7) "(i)n the standard approach, the terms 'utility maximization' and 'choice' are synonymous."

Suppose further that we are now told that the "entity" in the black box is a sophisticated machine and not a "flesh-and-blood human being." Somehow, we feel that because the information of a machine selecting the alternatives was previously withheld from us, we got tricked into the conclusion that it maximized its welfare. We feel strange to realize that we had ascribed welfare to a machine.

Note that the information of a machine selecting alternatives is “non-choice evidence” in the sense that it can not be tested with evidence on the selected alternatives (a claim to be discussed in the next section). Gul and Pesendorfer (2008, p. 8) write that “(i)f an economist proposes a new theory based on non-choice evidence then either the new theory leads to novel behavioral predictions, in which case it can be tested with revealed preference evidence, or it does not, in which case the modification is vacuous. In standard economics, the testable implications of a theory are its content; once they are identified, the non-choice evidence that motivated a novel theory becomes irrelevant.” Indeed, Gul and Pesendorfer (2008, p. 35) claim that “(p)opulating economic models with ‘flesh-and-blood human beings’ was never the objective of economists.”

If we accept Gul and Pesendorfer’s claim that only choice-evidence is relevant, then we are forced to conclude that the machine maximized its welfare. This clashes with our intuition of what standard economics is about. I think it is fair to claim that for most of the economists it is simply absurd to admit machines as individuals in welfare analysis. In standard economics we ascribe welfare to humans but are reluctant to do so for machines. The fact that standard economists happily make the metaphysical assumption that whenever we consider economic behavior we take it to be human behavior simply demonstrates that standard economics takes non-choice evidence *meaningfully* into account.

Even the proponents of the doctrine of choice-evidence-only take non-choice evidence into account. Otherwise, it is hard to make sense of the motivation provided for instance in Gul and Pesendorfer’s (2001) “Temptation and Selfcontrol”. To illustrate the point, below is an excerpt from their motivation in which I replace any reference to an agent with the word ‘machine’: ‘Consider a machine who must decide what to eat for lunch. It may choose a vegetarian dish or a hamburger. In the morning, when the machine feels no hunger, it prefers the healthy, vegetarian dish but at lunchtime, it experiences a craving for a hamburger. To lessen the impact of its lunchtime craving, the machine may seek to limit the options at lunchtime. For example, it may choose a vegetarian restaurant. When this is not possible and the machine is confronted with a menu that includes both the vegetarian meal and the hamburger, it may exercise self-control, that is, resist the craving for the hamburger and choose the vegetarian meal. Machines will frequently use both remedies. In the example, the machine may visit a vegetarian restaurant to exclude the hamburger from the option set. However, even the vegetarian restaurant offers an unhealthy dessert and self-control may be used to resist that temptation.’ Hungry salad

eating machines craving for hamburgers but exerting selfcontrol simply sound absurd. The altered motivation qualifies as science fiction but not as serious standard economics. Human beings rather than machines must have been the motivation. To be fair, Gul and Pesendorfer (2008, p. 8) acknowledge non-choice evidence as a source for motivation when writing “(i)n standard economics, the testable implications of a theory are its content; once they are identified, the non-choice evidence that motivated a novel theory becomes irrelevant.” We show, however, that the assumption of a human being rests on non-choice evidence. That is, standard economics has always content that can not be identified with choices. So non-choice evidence is not just used for motivations of theories.

At a second glance, our argument offered sounds rather silly. Isn't it just an intellectual exercise without relevance to standard economics if we bother the theory with the task of distinguishing between human beings and machines? I think that human economic behavior is precisely what lies beneath the enterprize of behavioral economics: By considering topics like fairness, altruism, temptation, selfcontrol, addiction, awareness, emotions etc., economists strive to make the homo oeconomicus more human-like. So I believe it is justified to ask what methodological constraints on this endeavor are imposed by the theory of choice. Or to put it differently, can behavioral economics be a purely behavioral theory?

The reader may find my arguments not relevant for two other reasons. First, he may believe that human beings can adequately be modeled as machines, or second, that machines should be ascribed welfare too. In either case, I will show that the theory of choice is incomplete in that the class of machines to which we can ascribe welfare to can not be well defined by any general procedure. Moreover, I argue that a theory capable of non-arbitrarily deciding which machines to ascribe welfare to must use information about decision processes and not just choices themselves.

3 Revealed Humans? - An Example

My argument in the previous section depends crucially on the assertion that the information that a machine selected the alternative is “non-choice evidence” in the sense that it can not be tested with evidence on the selected alternatives. The extent to which this assertion is true is discussed with an example in this section.

While machines (like a calculator) can perform standard calculations much better than humans, they also have certain limitations. Could those limitations be used to

reveal that a selection among alternatives is made by a machine rather than a human being?

Consider the following situation: Bob, an economist, meets Ann. He really loves her at first sight but before he proposes marriage to her, he wants to make sure that Ann is a human being. Ann looks like a beautiful human girl but since Bob is indoctrinated by Gul and Pesendorfer, he only accepts choice-evidence. That is, he wants to find out whether Ann is a human being or a machine by her/its choices alone.

He considers to pose to her the following choice problem: There are two alternatives, x and y . The default alternative is y . That is, unless Ann explicitly chooses x , she/it gets y . x pays 100 dollars and y pays 0 dollars in case there is no odd natural number that is the sum of two even natural numbers. Otherwise, if there is such a natural number, x pays 0 dollars and y pays 100 dollars.

Bob does not know whether Ann is a human being or a machine M_0 operating in the following way: It checks each natural number starting with 0, to see whether it is odd or not. If it is odd, then it checks whether it is the sum of two even natural numbers. To simplify, we assume that for each odd natural number it only checks lower even numbers. So for each odd number, it has to check only finitely many numbers. The computation halts only if it found an odd natural number that is the sum of two even natural numbers. If the machine halts, then it says “Yes” to alternative x .

Clearly, the machine Ann will never stop and will stick with alternative y , while any educated human Ann has no difficulty seeing that there can not be an odd natural number that is the sum of two even natural numbers, and hence with some taste for money she will choose x . So in this case, if Bob sees Ann sticking to y , it reveals she is not an educated human being but the machine.

In this example, Bob designed the choice problem so to reveal whether the choice is made by the human Ann or the *specific* machine Ann M_0 . In general, Ann could not just be this specific machine but *some type* of machine. In particular, there could be another machine Ann that is behaviorally equivalent to the human Ann for this choice problem. Such a machine does not need to be as sophisticated as the human Ann may be. For instance, consider the trivial machine M_{Yes} that always says “Yes” on any input. While it may be very convenient to have his future wife always say “Yes”, Bob may still prefer a human wife. But with the choice problem on hand he is unable to discriminate human Ann from “Yes”-machine Ann. To discriminate between this trivial “Yes”-machine Ann and human Ann, he would have to design another choice problem.

For instance, the problem could be “reversed” such that it pays 0 dollars on x and 100 dollars on y in case there is no odd natural number that is the sum of two even natural numbers. Otherwise, in case there is such a natural number, x pays 100 dollars and y pays 0 dollars. Now with this choice problem, Bob is able to discriminate between “Yes”-machine Ann and human Ann, but there may be yet another machine that in this choice problem is behaviorally equivalent to the human Ann. For instance, consider the machine that always selects the alternative that pays 100 dollars in the first case of any decision problem. In both decision problems so far discussed, such a machine Ann would be behaviorally equivalent to human Ann. So again, Bob needs to find a suitable choice problem to discriminate between such a machine Ann from the human Ann, which again may lead to choice-evidence that is behaviorally equivalent to some other machines, etc.

4 Some Impossibility Claims

The previous discussion begs the question of whether we can design a general choice procedure to reveal that a decision maker is human or a machine. Such a procedure would invalidate my argument against the Gul and Pesendorfer’s doctrine of admitting only choice-evidence since we could reveal whether or not a black box is a human being simply by choices. Unfortunately, we will show the following claims:

There is no choice procedure to decide whether or not a black box is a human being or a machine.

The claim will follow almost trivially from the fact that in finite choice problems given the choices of a human being, we could design a machine that exactly behaves as the human being.

As mentioned at the end of Section 2, the reader may not care to distinguish between choices made by human beings and machines because he may believe that human being’s can be modeled as machines. Unfortunately, we will show that not only can we never find out from choices alone whether a black box is a human being or a machine, we can not even decide the subset of machines that are behaviorally equivalent to human beings.

Could the doctrine of choice-evidence-only succeed at least with regard to machines? This would comfort a reader who believes that some machines should be ascribed welfare too. Such a conviction may be a consequence of our first claim since based on choice-evidence alone, we are forced by the revealed preference doctrine to attribute welfare to

some machines. Is there at least a general procedure to decide for each machine based on choice-evidence alone whether or not we have to attribute welfare to it?

There is no general choice procedure to decide for every machine whether or not it has some preference relation.

The claims follow from a general undecidability result:

Given a list of consistent axioms on choice behavior, there is no general procedure to decide for every machine whether or not the choices of the machine satisfy the list of axioms.

That is, the doctrine of admitting choice-evidence-only has limits even if applied to machines only. If one finds the ability to discriminate between machines and human beings irrelevant or believes that human beings are in some sense equivalent to machines, then one is still faced with the non-existence of a general procedure that could be given to the working economist to determine for each machine whether or not its choices satisfy a list of axioms.

To prove the claims, we need to make them precise by formalizing them. To this extent, we will make use of some definitions on Turing machines stated in the appendix. Our arguments will rely on the Church-Turing hypothesis according to which if there is any general procedure at all, a single set of instructions, given once and for all, that will allow us to decide the problems, then there must be a Turing machine which can carry out the general procedure. By general choice procedure we mean an algorithm that could be automatized to design and conduct choice experiments by working economists that takes as input choice problems and admits as output choices. With the methodological problem so formalized, it becomes a simple version of the *Halting Problem* that Alan Turing proved to be undecidable, a result closely related to Kurt Gödel's First Incompleteness Theorem.

5 Formal Statements

By the Church-Turing hypothesis, we take any machine to be a *Turing machine*. Any general procedure, recipe, computation or algorithm performed by a machine can be performed by a computation of a Turing machine. Turing machines can be thought of general purpose computers with an unlimited memory. In the appendix we provide

a formal definition of Turing machines (and further definitions used in this section). Briefly, a Turing machine functions as follows: Initially, a Turing machine M receives as input a string, a finite sequence of symbols from a finite input alphabet. This could be thought of as an infinite tape with the finite input string printed on the leftmost squares and the rest of the tape left being blank. The reading and writing head of the Turing machine starts in the starting state, denoted by q_0 , on the leftmost square of the tape. The computation of M proceeds according to the transition function, denoted by δ , to the next state, with the head moving either (L) left or (R) right, reading a finite number of symbols at a time on the tape, and eventually (over)writing symbols on the tape. If the head tries to move left off the tape, then the head stays at the same place for that move even though the transition function prescribes L. The computation continues until it moves to the accept state, denoted by q_a , or the reject state, denoted by q_r , at which points it *halts*. Otherwise, it *loops* forever. A Turing machine can be universal in the sense of taking as input descriptions of Turing machines as well.

Any object O encoded into a finite string of symbols is denoted by $\langle O \rangle$. Any finite mathematical object can be represented by a finite string. There are many different ways to encode objects into strings. Which way to choose does not matter since there is always a Turing machine that can translate one such encoding into another.

Consider a nonempty finite set of *alternatives* X . We restrict ourselves here to a finite set of alternatives because we think that any choice experiment can meaningfully involve in practice only a finite number of alternatives. A *choice problem* is given by a nonempty subset of alternatives $A \in 2^X \setminus \{\emptyset\}$. The interpretation is that A is the set of alternatives among which the decision maker has to choose. His/her/its choice is given by a *choice function* $c : 2^X \setminus \{\emptyset\} \longrightarrow 2^X \setminus \{\emptyset\}$ such that for all $A \in 2^X \setminus \{\emptyset\}$, $c(A) \subseteq A$.

We encode any choice problem with the nonempty set of alternatives A by a string denoted by $\langle A \rangle$, a finite sequence of symbols from the input alphabet of the Turing machine to which the choice problem is posed. Similarly, any Turing machine M can be encoded into a string denoted by $\langle M \rangle$.

Fix a choice function c . We say that a Turing machine M *chooses according to the choice function* c if for every $A \in 2^X \setminus \{\emptyset\}$ it accepts the input string $\langle A \rangle$ and outputs just the string $\langle c(A) \rangle$ on its tape.

Observe that if c is a choice function, then we can construct the Turing machine M_c that chooses according to the choice function c . (We use here a notation of Turing machines familiar from Sipser, 2006.)

$M_c =$ “On input string $\langle A \rangle$ for $A \in 2^X \setminus \{\emptyset\}$,

1. accept $\langle A \rangle$,
2. just output $\langle c(A) \rangle$.”

Remark 1 *If a human being’s choice is characterized by the choice function c , then there is a machine that is behaviorally equivalent to the human being’s choice, i.e., whose choice is also characterized by c . This follows from the construction of the Turing machine M_c . Hence, there is no procedure, even no special procedure tailor made for a particular choice function only, to decide based on choice-evidence alone whether a black box is a human being or a machine.*

Could the doctrine of choice-evidence-only succeed at least with regard to machines? To answer this question we seek general procedures that could decide based on choices for every machine whether or not its choices satisfy certain properties

In the theory of choice, properties of choice functions are captured by lists of “axioms”. A *list of axioms* \mathcal{P} is a subset of choice functions. We say that a list of axioms \mathcal{P} is *consistent* if \mathcal{P} is nonempty, i.e., there exists a choice function c that satisfies the list of axioms. We say that the *choices of a Turing machine M satisfy a list of consistent axioms \mathcal{P}* if M chooses according to a choice function c with $c \in \mathcal{P}$.

We seek a general choice procedure that allows us for any Turing machine to decide whether it’s choices satisfy a list of axioms \mathcal{P} . Formally, fix a list of consistent axioms \mathcal{P} . Consider the language

$$L_{\mathcal{P}} = \{\langle M \rangle : M \text{ is a Turing machine and it's choices satisfy the list of axioms } \mathcal{P}\}.$$

The language $L_{\mathcal{P}}$ is decidable if some Turing machine M decides the language, i.e., if M accepts any string in $L_{\mathcal{P}}$ and rejects any string not in $L_{\mathcal{P}}$. That is, if $L_{\mathcal{P}}$ is undecidable, then there exists no procedure - including no *general choice procedure* - that decides for any Turing machine whether it’s choices satisfy a list of axioms \mathcal{P} .

Theorem 1 *For any list of consistent axioms \mathcal{P} , the language $L_{\mathcal{P}}$ is undecidable.*

PROOF. The proof is by contradiction. We assume that $L_{\mathcal{P}}$ is decidable and use this assumption to show that the Halting problem H (see the appendix) is decidable. Since we know from Turing’s Theorem (Theorem 2 in the appendix) that H is undecidable,

we derive a contradiction. The key is to show that H is reducible to $L_{\mathcal{P}}$, and then use Theorem 3 in the appendix.

To demonstrate the mapping reducibility from H to $L_{\mathcal{P}}$ (see the appendix for a definition), we must present a computable function f such that

$$\langle M, \omega \rangle \in H \text{ if and only if } f(\langle M, \omega \rangle) \in L_{\mathcal{P}}.$$

The following Turing machine F computes a reduction f .

$F =$ “On input string $\langle M, \omega \rangle$:

1. Construct a Turing machine M' .
 $M' =$ “On input string $x \in \Sigma^*$:

 1. Run M on ω .
 2. If M halts on ω , then run M_c on x .”

2. Output $\langle M' \rangle$.”

If F determines that the input is not of the form as specified in line “On input string $\langle M, \omega \rangle$:”, then F outputs an arbitrary string not in $L_{\mathcal{P}}$.

Since \mathcal{P} is consistent, there is a choice function $c \in \mathcal{P}$. Let M_c in the description of F be defined as in front of Remark 1 using a choice function $c \in \mathcal{P}$. If $c \in \mathcal{P}$, then the choices of M_c satisfy the list of axioms \mathcal{P} .

If $\langle M, \omega \rangle \in H$ then M halts on ω . Thus F runs M_c and $\langle M' \rangle \in L_{\mathcal{P}}$. Conversely, if for $\langle M' \rangle \in L_{\mathcal{P}}$ with $\langle M' \rangle = f(\langle M, \omega \rangle)$ for some Turing machine M and some input string ω , we must have that $\langle M, \omega \rangle \in H$ since M must have halted on ω .

Since H is undecidable by Theorem 2 (see appendix), we have by Theorem 3 (see appendix) that $L_{\mathcal{P}}$ is undecidable. □

Theorem 1 shows that there is no general procedure with which the applied economist could decide whether or not a machine satisfies a list of axioms. There is an immediate special case of the theorem. Consider the following language

$$L_c = \{ \langle M \rangle : M \text{ is a Turing machine and chooses according to the choice function } c \}.$$

Corollary 1 *Suppose a human being’s choice function is given by c . Then the language L_c is undecidable.*

Thus, if a human being's choice is characterized by a choice function c , then the human being is not only behaviorally equivalent to a machine (Remark 1) but there is even no general procedure to decide the set of Turing machines that are behaviorally equivalent to this human being. In this sense, the set of machines that “models” a human being's behavior is not well-defined.

Given that based on choice-evidence we are forced to attribute welfare to some machines, is there at least a general procedure to decide for each machine whether or not we have to attribute welfare to it? To answer this question we need to consider preferences. A binary relation \succ on X is *asymmetric* if $x \succ y$ then $y \not\succeq x$, for $x, y \in X$. A binary relation \succ on X is *negatively transitive* if $x \not\succeq y$ and $y \not\succeq z$ implies that $x \not\succeq z$, for $x, y, z \in X$. We say that a binary relation \succ on X is a *preference relation* if it is asymmetric and negatively transitive.

Define a function $c_\succ : 2^X \setminus \{\emptyset\} \longrightarrow 2^X \setminus \{\emptyset\}$ by

$$c_\succ(A) = \{x \in A : \text{for all } y \in A, y \not\succeq x\} \text{ for all } A \in 2^X \setminus \{\emptyset\}.$$

Consider *Houthakker's axiom*: If $x, y \in A$, $x, y \in B$, and if $x \in c(A)$ and $y \in c(B)$, then $x \in c(B)$. There exists a preference relation \succ such that $c = c_\succ$ if and only if the choice function c satisfies Houthakker's axiom (Kreps, 1988, pp. 14-15).

We say that a Turing machine M *reveals a preference relation* if it chooses according to a choice function c that satisfies Houthakker's axiom.

Define the language

$$L_W = \{\langle M \rangle : M \text{ is a Turing machine and reveals a preference relation}\}.$$

Corollary 2 *The language L_W is undecidable.*

Thus, there is no general procedure to decide for each machine based on choices alone whether we have to attribute welfare to it or not.

The negative results of Theorem 1 and Corollaries 1 and 2 rest on the fact that machines could loop. Why should such results be of practical relevance? In any practical choice experiment, wouldn't we impose simply a deadline on the maximal time allowed to make choices?¹

¹This is what we did implicitly in Section 3 in which we specified y to be the default alternative.

Let M be a Turing machine that halts on all inputs. The running time of M is the function $t : \mathbb{N} \rightarrow \mathbb{N}$, where $t(n)$ is the maximum number of steps that M uses on any input length n .

A Turing machine M chooses according to the choice function c *within time* m if for every $A \in 2^X \setminus \{\emptyset\}$ it accepts the input string $\langle A \rangle$ and outputs just the string $\langle c(A) \rangle$ on its tape in at most m steps, i.e. $t(\langle A \rangle) \leq m$ for all $A \in 2^X \setminus \{\emptyset\}$.

Define the language

$$L_{c,m} = \left\{ \langle M \rangle : \begin{array}{l} M \text{ is a Turing machine and it chooses according} \\ \text{to the choice function } c \text{ within time } m \end{array} \right\}.$$

Proposition 1 *For every choice function c and any finite time m , the language $L_{c,m}$ is decidable.*

PROOF. Since X is finite, we can enumerate all $A \in 2^X \setminus \{\emptyset\}$. Let A_1, \dots, A_k be the list of all nonempty subsets of X . Fix a choice function $c : 2^X \setminus \{\emptyset\} \rightarrow 2^X \setminus \{\emptyset\}$ and a finite time m .

Consider the following Turing machine:

$D =$ “On input string $\langle M \rangle$: Set $\ell = 1$.

1. If $\ell > k$, accept. Otherwise, run M on $\langle A_\ell \rangle$.
2. If M accepts $\langle A_\ell \rangle$ and outputs $\langle c(A_\ell) \rangle$ on its tape in at most m steps, then go to 1. setting ℓ to $\ell + 1$. Otherwise, reject.”

If D determines that the input is not of the form as specified in line “On input string $\langle M \rangle$:”, then D rejects.

Clearly, for every input the Turing machine D accepts or rejects. Moreover, it accepts if and only if M chooses according to c within time m . Thus, D is a decider of the language $L_{c,m}$. \square

Proposition 1 means that imposing a finite running time solves the problem. That is, for every finite running time we can decide for every Turing machine whether or not it chooses according to a choice function c or not.

Unfortunately, I never came across a theory of choice that provides such a running time. This may be due to the fact that choice-theorists simply assume that any choice

is human choice and do not seriously consider choices by machines. Despite the fact that for every finite running time, there is a general procedure that allows us to decide for every machine whether or not it behaves according to a choice function c , the set of entities to which the theory of choice applies remains arbitrary without a theory of running times. This is because some machines may choose according to c for running time m although they don't for running time $m' < m$. So in this sense the theory of choice remains "incomplete" even if applied to machines only.

What is a reasonable finite running time? It is hard to imagine a theory of running times that does not take into account the *process of choice* (besides a measure of the size of the choice problem). Unfortunately, the standard theory of choice is mostly silent on the internal processes of how the black box arrives at certain choices.

6 Discussion

6.1 Conclusions

Our results give rise to the following conclusion:

If standard economics is based solely on the theory of choice, then it must be "incomplete" in the sense that it has content that is undecidable by any general choice procedure.

We have shown that in finite decision problems, human choice is behaviorally indistinguishable to choices by machines (Remark 1). Thus, choice-evidence-only does not allow us to test whether choices are made by humans or machines. Even if one models human choice by machines or admits machines to welfare analysis, then the class of machines that can be ascribed welfare to is undecidable (Corollary 2). A criterion that would make the problem decidable (Proposition 1) - like procedural features - are not provided by the theory of choice.

Standard economists are not bothered by those problems. This is because applied economists happily make implicit assumptions about decision makers. That is, standard economics is not based on choice-evidence-only as claimed by Gul and Pesendorfer (2008). A doctrine admitting choice-evidence-only is counter-productive for the further development of economics since it essentially forbids the scientific analysis of current

metaphysical assumptions in economics that can not be revealed by choices alone. We should not fool ourselves in having achieved already a logical positivist foundation of standard economics. Nor can we expect the theory of choice alone to deliver a pure logical positivist foundation. Other sources than choice data may be used for the content that can not be revealed with choices. For example, one important source that distinguishes social sciences from natural sciences is *introspection*. Often interpretations of theories of choice stem from intuitions gather through personal introspection. Can we make introspection amenable to scientific analysis?

The argument put forward in this note shall not be taken as directed against the use of choice-evidence in standard economics. Rather, it is against the doctrine of admitting *only* choice-evidence in standard economics. Moreover, we show that there is no general procedure (not just no general choice procedure) that allows us to decide on machines satisfying a list of consistent axioms \mathcal{P} . So even if general procedures are allowed to open the black box (i.e. able to access the description of the machines), then they are unable to decide for each machine whether or not it satisfies the list of axioms \mathcal{P} . This suggests that there are limits to general procedures, not just general choice procedures, at least when it comes to machines.

In Section 3, Bob faces a paradox: Being a follower of the choice-evidence-only doctrine suggests to him that if robot Ann's choices are indistinguishable from human Ann's choices, then he should happily get married with robot Ann (e.g. as illustrated in "Love and Sex with Robots" by David Levy, 2007). On the other hand, he is faced with his desire for a human Ann. This is just a fable for the position in which Gul and Pesendorfer may put the standard economist.

6.2 Caveats

First, I do not claim that it is impossible to discriminate between a human being and a particular machine based on choice-evidence. I presented counter examples in Section 3. Rather I claim that for any finite choice problem and choices by a human being, there is a behaviorally equivalent machine (Remark 1).

Second, I do not claim that it is impossible to discriminate between humans and machines. I just claim that there is no choice procedure that allows us to decide whether the black box is a human being or a machine. Other means may be readily available. For instance, Bob could put Ann into the brain scanner and immediately see whether she

is a flesh-and-blood human being or a machine. So clearly there may be very effective procedures to discriminate between human beings and machines outside the realm of choice procedures.

Third, Theorem 1 means that it is impossible for a general procedure to solve for *each* machine whether or not it satisfies a list of axiom \mathcal{P} . It does not mean that for a *given* machine M we must have no way to find out whether it satisfies a list of axioms \mathcal{P} .

Fourth, contrary to a first glance, Corollary 1 does not say anything about human behavior except that the human's choice is assumed to be characterized by a choice function.

Fifth, the fact that there exist general procedure to decide for every machine whether it chooses according to a choice function c within finite time m does not imply that such a general procedure is computationally feasible in practice. Results on how fast the running time of a general procedure for checking certain choice axioms would grow in the size of choice problem may depend on the nature of the choice axioms.

6.3 Related Literature

Others have questioned the choice-evidence-only doctrine before me. A critical debate of Gul and Pesendorfer (2008) paper is collected in Caplin and Schotter (2008). For instance, Schotter (2008, p. 72) mentions that “it is only when choice is consistent not only with predictions of a theory but also with the reasons stipulated for that choice that we can be confident that the predictions of the theory will remain valid when the parameters of the model change. ... Knowing why a choice was made may require data other than choice.”

I should acknowledge some related ideas in the philosophical literature. First, in Section 3, the idea of a machine Ann that is behaviorally undistinguishable from a human Ann is essentially equivalent to the Zombie argument discussed extensively in the literature on consciousness. Searle (1997, p. 146) summarizes the argument: “If it is logically possible, in the sense of being not self-contradictory, to imagine that there could be zombies that were organized just as we are and had exactly our behavior patterns, but were totally devoid of consciousness, then it follows that our consciousness cannot logically consist simply in our behavior or functional organization.” Second, the choice problem to distinguish human Ann from the particular machine Ann M_0 in Section 3 was inspired by Penrose (1994, Section 2). He shows more generally that for every Turing

machine there is a procedure that humans can analyze but the machine can not. This, of course, does not mean that another machine could't analyze the procedure as well.

Finally, I should acknowledge a related but different literature which asks about the computability of choice functions (e.g. Lewis, 1985), i.e., whether there is a Turing machine that computes a choice function. In this literature, Turing machines are seen as ideal generalizations of human cognitive abilities, and non-computability of a choice function (on an infinite domain) is interpreted as a form of “computational irrationality”. Different from that literature, we are not concerned with the computability of a choice function but rather with the methodological question about general choice procedures that allow us to reveal for any black box whether or not its choices satisfy certain properties. Moreover, we are skeptical about the interpretation of Turing machines as ideal generalizations of human cognitive capabilities.

A Turing Machines

In this appendix, we collect some definitions on Turing machines that are used in the formal statement of the problem. For further details see Sipser (2006).

A *Turing machine* is a tuple $(Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$, where

1. Q is a finite set of *states*,
2. Σ is a finite *input alphabet* not containing the *blank symbol* \sqcup ,
3. Γ is the finite *tape alphabet*, where $\sqcup \in \Gamma$ and $\Sigma \subseteq \Gamma$,
4. $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$ is the *transition function*,
5. $q_0 \in Q$ is the *start state*,
6. $q_a \in Q$ is the *accept state*, and
7. $q_r \in Q$ is the *reject state*, where $q_r \neq q_a$.

Initially, a Turing machine M receives as input a string $\omega \in \Sigma^*$, where Σ^* is the set of all finite sequences of symbols from the finite input alphabet Σ of the Turing machine M . This could be thought of an infinite tape with the finite input string ω printed on the leftmost squares and the rest being blank. The reading and writing head of the Turing

machine starts in start state q_0 on the leftmost square of the tape. The computation of M proceeds according to the transition function δ to the next state, with the head moving either (L) left or (R) right, reading a finite number of symbols at a time on the tape, and eventually (over)writing symbols on the tape. If the head tries to move left off the tape, then the head stays at the same place for that move even though the transition function prescribes L . The computation continues until it moves to the accept state q_a or the reject state q_r at which point it halts. Otherwise, it *loops* forever.

A *configuration* $C = \omega q_i \nu$ of a Turing machine M consists of two strings ω and ν over the tape alphabet Γ and a state $q_i \in Q$ such that at state q_i the head is on the leftmost square of the string ν , and to the left the string ω is written on the tape.

Let $\omega, \nu \in \Gamma^*$ and $a, b, c \in \Gamma$. A configuration $C_1 = \omega a q_i b \nu$ yields a configuration $C_2 = \omega q_j a c \nu$ if $\delta(q_i, b) = (q_j, c, L)$ (leftward move). Similarly, a configuration $C_1 = \omega a q_i b \nu$ yields a configuration $C_2 = \omega a c q_j \nu$ if $\delta(q_i, b) = (q_j, c, R)$ (rightward move). We can define the transitive closure $C_1 \rightarrow C_n$ if there exist configurations C_1, \dots, C_n such that C_ℓ yields configuration $C_{\ell+1}$ for $\ell = 1, \dots, n-1$.

A Turing machine M *accepts* input string ω if there exist configurations $C_1 = q_0 \omega$ and C_n such that $C_1 \rightarrow C_n$ and C_n is an accepting configuration, i.e., a configuration with state q_a .

A collection of strings is a language. The collection of strings accepted by the Turing machine M is the *language* of M , denoted by $L(M)$. A language is *acceptable* (or Turing-recognizable or recursively enumerable) if some Turing machine accepts it.

A Turing machine M *rejects* input string ω if there exist configurations $C_1 = q_0 \omega$ and C_n such that $C_1 \rightarrow C_n$ and C_n is a rejecting configuration, i.e., a configuration with state q_r .

A Turing machine M *halts* on input string ω if it either accepts or rejects ω .

A Turing machine M *decides* a language L if M accepts any string $\omega \in L$ and rejects any string $\omega \notin L$. In this case, M is called a decider of L .

A language is *decidable* (or recursive) if some Turing machine decides it.

Recall from Section 5 that an object O encoded into a finite string of symbols is denoted by $\langle O \rangle$.

The Halting Problem is now formalized as follows:

$$H = \{ \langle M, \omega \rangle : M \text{ is a Turing machine and } M \text{ halts on input string } \omega \}.$$

Theorem 2 (Turing, 1936) *H is undecidable.*

Many undecidable problems are proved by *reduction* of the Halting Problem to the problem of interest. Reducibility is formalized as follows:

A function $f : \Sigma^* \rightarrow \Sigma^*$ is a *computable function* if some Turing machine M , on every input ω , halts with just $f(\omega)$ on its tape.

The language A is *mapping reducible* to language B if there is a computable function $f : \Sigma^* \rightarrow \Sigma^*$, where for every ω ,

$$\omega \in A \text{ if and only if } f(\omega) \in B.$$

The function f is called a reduction of A to B .

Theorem 3 *If language A is mapping reducible to language B and A is undecidable, then B is undecidable.*

For a proof see Sipser (2006, Theorem 5.22 and Corollary 5.23).

References

- [1] Caplin, A. and A. Schotter (eds.) (2008). *The foundations of positive and normative economics: A handbook*, Oxford University Press.
- [2] Gul, F. and W. Pesendorfer (2008). The case for mindless economics, in: Caplin, A. and A. Schotter (eds.), 3-39.
- [3] Gul, F. and W. Pesendorfer (2001). Temptation and selfcontrol, *Econometrica* 69, 1403-1435.
- [4] Kreps, D.M. (1988). *Notes on the theory of choice*, Westview Press.
- [5] Levy, D. (2007). *Love and sex with robots. The evolution human-robot relationships*, HarperCollins Publishers.
- [6] Lewis, A.A. (1985). On effectively computable realizations of choice functions, *Mathematical Social Sciences* 10, 43-80.

- [7] Penrose, R. (1994). *Shadows of the mind. A search for the missing science of consciousness*, Oxford University Press.
- [8] Schotter, A. (2008). What's so informative about choice?, in: Caplin, A. and A. Schotter (eds.), 70-94.
- [9] Searle, J.R. (1997). *The mystery of consciousness*, New York Review Books.
- [10] Sipser, M. (2006). *Introduction to the theory of computation*, 2nd edition, Course Technology.
- [11] Turing, A. (1936). On computable numbers, with an application to the Entscheidungsproblem, *Proceedings of the London Mathematical Society Series II*, 42, 230-265.