Journal of Mathematical Economics 66 (2016) 26–39

Contents lists available at ScienceDirect

Journal of Mathematical Economics

journal homepage: www.elsevier.com/locate/jmateco

An introduction to mechanized reasoning[☆]

Manfred Kerber[a], Christoph Lange[b], Colin Rowat[c,*]

[a] School of Computer Science, University of Birmingham, UK
[b] Fraunhofer IAIS and University of Bonn, Germany
[c] Department of Economics, University of Birmingham, Edgbaston B15 2TT, UK

ARTICLE INFO

Article history:
Received 12 January 2015
Received in revised form
22 June 2016
Accepted 23 June 2016
Available online 18 July 2016

Keywords:
Mechanized reasoning
Formal methods
Social choice theory
Auction theory

ABSTRACT

Mechanized reasoning uses computers to verify proofs and to help discover new theorems. Computer scientists have applied mechanized reasoning to economic problems but – to date – this work has not yet been properly presented in economics journals. We introduce mechanized reasoning to economists in three ways. First, we introduce mechanized reasoning in general, describing both the techniques and their successful applications. Second, we explain how mechanized reasoning has been applied to economic problems, concentrating on the two domains that have attracted the most attention: social choice theory and auction theory. Finally, we present a detailed example of mechanized reasoning in practice by means of a proof of Vickrey's familiar theorem on second-price auctions.

1. Introduction

Mechanized reasoners automate logical operations, extending the scope of mechanical support for human reasoning beyond numerical computations (such as those carried out by a calculator) and symbolic calculations (such as those carried out by a computer algebra system). Such reasoners may be used to formulate new conjectures, check existing proofs, formally encode knowledge, or even prove new results. The idea of mechanizing reasoning dates back at least to Leibniz (1686), who envisaged a machine which could compute the validity of arguments and the truth of mathematical statements. The development of formal logic from 1850 to 1930, the advent of the computer, and the inception of *artificial intelligence* (AI) as a research field at the Dartmouth Workshop in 1956 all paved the way for the first mechanized reasoners in the 1950s and 1960s.[1]

Since then, mechanized reasoning has been both less and more successful than anticipated. In pure maths, mechanized reasoning has helped prove only a few high-profile theorems. Perhaps surprisingly – although consistent with the greater success of applied AI over 'pure' AI – mechanized reasoning and formal methods[2] have enjoyed greater success in industrial applications, as applied to both hardware and software design. In the past

☆ We are grateful to Makarius Wenzel for help refining our code, to Marco Caminati for research assistance, to Peter Cramton, Paul Klemperer, Peter Postl, Indra Ray, Rajiv Sarin, Arunava Sen and Ron Smith for comments, and to the EPSRC for funding (EP/J007498/1). Rowat thanks Birkbeck for its hospitality. The presentation of the formal proof of Vickrey's theorem is based on Kerber et al. (2014). Finally, we are grateful to two anonymous referees and the co-editor for working with us to improve this paper.

* Corresponding author.
E-mail address: c.rowat@bham.ac.uk (C. Rowat).

1 Perhaps unsurprisingly, Gardner was ahead of his time in mechanized reasoning as well: four years before his regular columns with *Scientific American* began, his first article for them included a template allowing readers to make their own mechanized reasoners—out of paper.

2 The term *formal methods* is used here to denote approaches to establishing the correctness of mathematical statements to a precision that they can be meticulously checked by a computer. Rather than being seen as distinct from other mathematical methods, researchers in the area see them as the next step in mathematics' march towards greater precision and rigour (Wiedijk, 2008). Consider: "A Mathematical proof is rigorous when it is (or could be) written out in the first-order predicate language $L(\in)$ as a sequence of inferences from the axioms ZFC" (MacLane, 1986). The advantages of taking this next step with computers include: a computer system is never tired or intimidated by authority, it does not make hidden assumptions, and can easily be rerun. A pioneer of mechanized reasoning – who saw himself building on Bourbaki's formalism – referred to computers as "slaves which are such persistent plodders" (Wang, 1960).

http://dx.doi.org/10.1016/j.jmateco.2016.06.005

decade or so, computer scientists have also begun to apply formal methods to economics.

A central inspiration for this recent work are Geanakoplos' three brief proofs of Arrow's impossibility theorem (Geanakoplos, 2005).[3] Initially, Nipkow (2009), Wiedijk (2007), and Wiedijk (2009) used theorem provers to encode and verify two of Geanakoplos' proofs. A subsequent generation of work, drawing on the inductive proof of Arrow's theorem in Suzumura (2000), used formal methods to discover new theorems. Tang and Lin (2009) introduced a hybrid technique, using computational exhaustion to show that Arrow holds on a small base case of two agents and three alternatives, and then manual induction to extend that to the full theorem. By inspecting the results of the computational step, they were able to discover a new theorem subsuming Arrow's. Tang and Lin (2011a) used this approach – exhaustively generating and evaluating base cases, and then using a manual induction proof to generalize the results – to establish uniqueness conditions for pure strategy Nash equilibrium payoffs in two player static games; they published manual proofs of two of the most significant theorems discovered this way in Tang and Lin (2011b). Geist and Endriss (2011) used the approach to generate 84 impossibility theorems in the 'ranking sets of objects' problem (Barberà et al., 2004).

To date, the economics literature remains almost untouched by research applying mechanized reasoning to economic problems.[4] The one exception that we are aware of is Tang and Lin (2011b), whose two theorems were discovered computationally, but proved manually.[5] As it is our view that these tools will become increasingly capable, this paper aims to introduce economists to mechanized reasoning.[6] It does so by means of three analytical lenses, each with narrower scope but greater magnification than its predecessor.

First, Section 2 presents an overview of mechanized reasoning in general. We do so by setting out a classificatory scheme, with the caveat that it should not be seen as implying a partition on the field: interesting research will straddle boundaries, perhaps even forcing them to be redefined.[7]

Second, Section 3 surveys the emerging literature applying mechanized reasoning to economics. We structure this survey primarily according to the problem domain within economics, referring only secondarily to our classificatory scheme. We do this to focus on the economic insights – primarily within social choice and auction theory – made possible by these techniques, rather than on the techniques *per se*.

Finally, to make this introduction more concrete, Section 4 provides an example of what mechanized reasoning looks like in practice, presenting a blueprint of a mechanized proof of Vickrey's theorem on second-price auctions. We present such an established theorem to focus attention on its implementation.

Section 5 concludes, and suggests some possible next steps for mechanized reasoning in economics.

2. Mechanized reasoning

Our overview of mechanized reasoning distinguishes between deductive and inductive systems. While the distinction has been recognized at least since Aristotle, deductive reasoning – which allows reliable inference of unknown facts from established facts – has been in the focus of the mechanized reasoning community. Inductive reasoning also generalizes from individual cases, but does not restrict itself to reliable inferences; the cost of this additional freedom is that its conjectures must then be tested.

2.1. Deductive reasoning

Historically, deductive reasoning systems were among the first AI systems, dating back to the 1950s. While the origins of deductive reasoning date to at least Aristotle, modern advances in this area built on the work of logicians in the second half of the 19th century and the start of the 20th (e.g. Whitehead and Russell, 1910). At the Dartmouth Workshop in 1956, Newell and Simon introduced the Logic Theorist, an automated reasoner which re-proved 38 of the 52 theorems in chapter of Whitehead and Russell's *Principia Mathematica* (Whitehead and Russell, 1910).[8]

Abstractly, a deductive reasoner implements a *logic* – which is comprised of a *syntax* defining well-formed formulae and a *semantics* assigning meaning to formulae – and a *calculus* for deriving formulae (called theorems) from formulae (called premises or axioms). Historically, subfields of mechanized reasoning have been defined by choice of logic, calculus and problem domain. This section provides a classificatory scheme based, first, on the choice of calculus. Following the choice of calculus, a logic is chosen to balance expressiveness and tractability. Finally, the problem domain itself will dictate some of the specialized features of a mechanized reasoner.

When a mechanized reasoner applies the calculus' permissible operations to the axioms to obtain new, syntactically-correct formulae it does not make use of the semantics: the semantics, or ascribed meanings, yield models that may assist human intuition, but which are not necessary to the formal process of reasoning itself.[9] Crucially, mechanized reasoning involves manipulating symbols.[10]

Thus, mechanized deductive reasoning since the Logic Theorist has seen reasoning as a search task for a syntactically well-defined goal.[11] Further, as the spaces through which search occurred was potentially large, successful reasoning would use *heuristics* to avoid unprofitable sequences of operations. From this point of view,

3 All three use Barberà's replacement of Arrow's *decisive voter* with a *pivotal voter* (Barberà, 1980). Barberà (1983) also used this approach to find a direct proof of the Gibbard–Satterthwaite theorem.

4 A recent symposium on economics and computer science, involving central figures at the interface between the disciplines, made no mention of mechanized reasoning (q.v. Blume et al., 2015).

5 The process by which the theorems were discovered is described in Tang and Lin (2011a,b) itself is all but silent on its mechanized origins.

6 For more general introductions, see Wiedijk (2008) and Avigad and Harrison (2014). Harrison (2007) introduces mechanized reasoning alongside computer algebra, presenting something of a unified view.

7 For example, we shall see that mechanized theorem discovery is usually associated with inductive reasoning. However – in economic examples – the most fruitful examples of theorem discovery (Tang and Lin, 2009, 2011a,b; Geist and Endriss, 2011) have combined very simple deductive reasoning systems with human intelligence.

8 According to McCorduck (2004), Russell himself "responded with delight" when shown the Logic Theorist's proof of the isosceles triangle theorem, whose proof was more elegant than their manual one.

9 Beginning with Euclid's efforts to axiomatize geometry, logicians have produced syntactical descriptions that make semantic references obsolete: Hilbert allegedly said that we would still have an axiomatization of geometry if we replaced the words 'point', 'line', and 'plane' by 'beer mug', 'bench', and 'table' (Hoffmann, 2013, p. 6).

10 That this was an insight at one point may be inferred from Turing's famous explanation that, "computing is normally done by writing certain symbols on paper" (Turing, 1936).

11 As noted by Harrison (2007), specialist provers have also been developed for particular problems for which more structured approaches than general search are appropriate.