Blueprint for an Algorithmic Economics*

Stephen Kinsella  
Department of Economics,  
Kemmy Business School,  
University of Limerick,  
Ireland  
stephen.kinsella@ul.ie

Received 17 October, 2008  
Revised

Algorithmic economics helps us stipulate, formulate, and resolve economic problems in a more precise manner than mainstream mathematical economics. It does so by aligning theorising about an economic problem with both the data generated by the real world and the computers used to manipulate that data. Theoretically coherent, numerically meaningful, and therefore policy relevant, answers to economic problems can be extrapolated more readily using algorithmic economics than present day mathematical economics. An algorithmic economics would allow mathematical economics to prove theorems relating to economic problems, such as the existence of equilibria defined on some metric space, with embedded mechanisms for getting to the equilibria of these problems. A blueprint for such an economics is given and discussed with an example.

Keywords: Algorithmic Economics; Mathematical Economics; Complexity Theory.

1. Searching for algorithmic content in modern macroeconomics

Why should we rid mathematical economics of its non-algorithmic content, and if it is a good idea to do so, how should we go about this removal? In attempting to answer these questions, Velupillai\(^2\) gives the reader an outline, or a blueprint, for an algorithmic economics. We see why, where, and how modern mathematical economics\(^3\) is replete with non-algorithmic content.

Velupillai’s article uses a series of examples\(^4\) to demonstrate the inherent dangers of non-algorithmic thinking for mathematical economists. Following a tour of

---

*My thanks to K. Vela Velupillai for his courage in forging a new research program in mathematical economics along constructive, computable, and algorithmic lines, as well as his support, mentoring, and stirring friendship throughout the years. My thanks to the editor of this journal for the opportunity to develop the ideas contained herein. In gratefully acknowledging his role in the development of this paper, I in no way implicate Prof. Velupillai in its errors.

\(^{2}\)Defined for the purposes of this article as general equilibrium theory, game theory, and recursive macroeconomics.

\(^{3}\)The intermediate value theorem, the Bolzano-Weierstrass theorem, and the Hahn-Banach theorem.
the history of the development of mathematical economics in parallel with the development of Bishop’s² approach to constructive analysis, and in comparison with Feynman’s path integral approach and Dirac’s development of the delta function, we see the bare bones of modern mathematical economics exposed as inherently non-algorithmic.

Modern mathematical economics, Velupillai argues, largely lacks numerical meaning, perhaps because unlike Feynmann’s vision of a mathematically meaningless but physically (and numerically) useful path integral approach, mathematical economics has no natural physical analogy to play against. The citadel of mathematical economics is based upon existence mathematics rather than the mathematics of the digital computer. Economic problems therefore must be cut to fit a classical-mathematical cloth, which is undesirable. As Velupillai, in this issue, puts it:

[t]he mathematical economist bends and moulds the economic concepts to fit the concepts and methods of classical mathematics, in the first stage. At a second stage, (illegitimate) approximations are attempted to compute demonstrably uncomputable and non-constructive entities.

So a problem exists within modern mathematical economics. We are not solving economic problems at the coalface of economic theory. We are solving mathematical problems which are then translated back into economic terminology for interpretation and extension by the profession. Velupillai argues the mapping from problem to mathematical definition to solution is not ideal, and suggests a better method of attack, which he calls algorithmic economics.

Algorithmic economics is the formalisation of economic theory on effective lines, where theorems about, functions on, estimation, and interpretation of economic phenomena are couched using the mathematics of the digital computer. The benefits of this approach are clearly spelled out in his ‘blueprint’ paper, and I reiterate and expand upon those benefits here.

We are also given a unique glimpse into how one might think about weakening and strengthening hypotheses and conclusions to tailor the theorems of mainstream mathematical economics to obtain approximately exact solutions in an example concerning the intermediate value theorem. We see that the data types, range and domains of the functions we build from the theorems we prove delimit and decide for us whether these functions are computable or not, and whether they will provide us with exact approximations when we test them against the real world. Velupillai’s demonstration is extremely valuable both intellectually and pedagogically, because it provides the reader with a blueprint for how one might go about doing algorithmic economics.

This paper summarises, expands upon, and critiques Velupillai’s contribution, mainly discussing the differences between approximate equilibria and perfect equilibria (section 2). The paper also makes a few short comments on the important differences between algorithmic, computable, and constructive mathematics (section
2. On actual versus approximate equilibria

What is the natural\(^a\), normal, and everyday computational situation in economics? We are given a set of data from which to draw inferences about the real world, using a model developed from economic theory, which relies on the theorems of mathematical economics. We go to a computer, and attempt to ‘fit’ the causal model we have developed to the data we have, and the computer gives the researcher the best fit it can, using the numerical analysis techniques at its disposal. Where are the pitfalls in this method, and what are the consequences of ignoring these pitfalls? Velupillai gives a partial answer in his paper.

Now think of a procedure to ascertain whether an equilibrium exists. Herbert Scarf\(^b\) gave a method to find an approximate equilibrium using simplicial subdivision, which, amongst others, launched the field of applied computable general equilibrium modeling\(^c\) (and it’s offshoot, modern policy-driven development economics), and recently several authors have generated results pertaining to the complexity of finding these approximate equilibria\(^d\). Economics is an applied, quantitative discipline at heart. We wish to say something concrete about the real world.

Economics has a long tradition of relying upon quantitative models both for presenting its theory and testing this theory empirically. Joseph Schumpeter\(^e\), writing in the first edition of *Econometrica* in 1933, described economics as the most quantitative of all sciences, including physics, because:

> [s]ome of the most fundamental economic facts, . . . , already present themselves as quantities made numerical by life itself. They carry meaning only by virtue of their numerical character.

But where, in modern mathematical economics, is the numerical character Schumpeter spoke of to be found?

Take the example of optimisation. In theoretical models, individual agents are presented as utility- or profit-maximisers. Selecting and estimating models for given data sets amounts, in the end, to optimisation—sums of squares are minimised and likelihoods are maximised on computers so routinely that often researchers may not even be aware that fitting a model means optimising it. In the normal course of their day, economists optimise unknowingly and without, I suspect, a strong sense of what it is they are doing using this procedure. The point of such ‘applied’ work is to obtain a numerical answer which can be interpreted by the researcher as the output of their study.

When building models, economists are often limited by the fact that the model later *needs* to be solved, and typically in a closed-form. Some researchers have aban-

\(^a\)Natural, in the sense of the quotidian character of life as a professional economists.

\(^b\)An approach shown recently by Velupillai\(^e\) to be formally uncomputable. The irrelevance of these models for policy has yet to be acknowledged in the mainstream literature.
Stephen Kinsella

doned relying on ‘representative agent’ models, and have opted for more complex models where a closed form solution is either unwieldy or impossible to generate, relying on computer simulations to obtain their results. Such agent-based models, if they are to be a viable alternative to more standard approaches, need to be ‘tuned’ so the results from these models coincide with observed empirical facts. This, again, is an estimation, and hence, an optimisation problem. The majority of researchers in applied economics, however, are firmly in the first camp: take data; regress it somehow; test those regressions; derive policy conclusions.

More formally, optimisation problems in estimation and modelling are typically expressed as max \( f(x) \), where \( f \) might be, say, a likelihood function, and \( x \) a matrix of the models parameters. The problem max \( f(x) \) is often considered synonymous with the solution matrix \( x^* \), following the application of a computational method such as numerical optimisation.

McCullough and Vinod enforce this point, writing that

\[ \text{any textbooks convey the impression that all one has to do is use a computer to solve the problem, the implicit and unwarranted assumptions being that the computers solution is accurate and that one software package is as good as any other...} \]

Yet even reviews of statistical software pay little attention to accuracy.

Accuracy is paramount in scientific work. Several studies have pointed out the rather egregious fact that many optimisation problems in econometrics cannot be solved with standard methods because of the existence of multiple optima and discontinuities, a subject Velupillai returns to frequently. Numerical optimisation problems can occur in widely used models, for example, generalised autoregressive conditional heteroscedasticity (GARCH) regressions. Another example comes from robust regressions using in testing growth and development theories. One can generate the objective function for estimating a linear regression by minimising the Least Median of Squares (LMS) instead of the Least Mean of Squares, and to show such a function has many local optima and does not appear very smooth.

These two examples illustrate different problems in optimisation. The GARCH model shows even widely applied models are not necessarily easy to estimate, and researchers are sometimes not even aware of the computational difficulties. The LMS model serves as an example where possibly superior models are not used because they seem difficult to estimate.

Standard methods, for example built on maximum likelihood estimation, are

\[ \text{Obviously numerically, max } f(x) \text{ is the same as min } f(x) \text{ in the models typically assumed in mainstream work.} \]

\[ \text{The problem of inaccurate software is not a 21st Century problem. Longley in 1967 discovered the same problems in the econometric softwares of his day, as McCullough and his colleagues discuss.} \]

\[ \text{Which tacitly assumes that a solution exists, might be stable, and is unique.} \]
essential hill-climbing algorithms. They begin with a given solution, and then proceed deterministically to the direction where the likelihood function’s value increases, for instance by following the direction of the gradient via the Hessian or bordered Hessian matrices, respectively. Such an algorithm is likely to fail in a situation with many local optima, since it will stop at the first peak it finds, typically\(^24\). Some methods also require the landscape, i.e. the objective function, to have certain properties such as continuity and be everywhere differentiable, amongst other assumptions. If these properties are missing, the algorithm performes does not work. What is worse, the employed software may not report an error but just return some value, taken in some sense as ‘true’ by the well-intentioned researcher simply searching for an answer.

Returning to the question posed at the beginning of this section, we see that economists, as optimisers, search through data for roots of equations, holding them to be equilibria as part of a grander non-algorithmic economic theory. In doing so, they rely on numerical methods to find these points. The numerical methods, the software, and the non-algorithmic theory produce errors at each stage of the estimation process. Velupillai’s work shows us a way to rectify, reduce, and cope with the three overlapping errors inherent in the standard approach.

This is part of the problem Velupillai shows us. Let us now think about ways of producing algorithmic content within modern mathematical economics in a positive direction. I take general equilibrium theory as my example.

2.1. **An example**

First, some notation\(^1\).

The general equilibrium problem is to discover prices and allocations of goods to individual traders such that each trader obtains the greatest possible utility. It has been shown\(^1\) that market clearing prices will exist, but, given the importance of the topic, relatively few algorithmic approaches to the discovery of equilibrium prices have been mooted\(^1\). The actual process of equilibrium price discovery is equivalent to finding fixed points with arbitrary initial conditions\(^19\).

Imagine \(m\) economic agents trading \(n\) goods. Let \(\mathbb{R}^N_+\) denote non negative coordinates inside \(\mathbb{R}^N\). The \(j\)th coordinate in \(\mathbb{R}^N\) represents an amount of good \(j\). Each trader \(i\) indexed by \(i = (1, \ldots, n)\) has a concave utility function \(u_i\) such that \(u_i : \mathbb{R}^N_+ \rightarrow \mathbb{R}^N\). This represents the trader’s preferences for different bundles of goods. Each trader has an endowment of goods, \(w_i = (w_{i1}, \ldots, w_{in}) \in \mathbb{R}^N_+\), so there is a \(y \in \mathbb{R}^N_+\) such that \(u_i(y) > u_i(x)\). Assuming monotonicity of \(u_i\) implies \(u_i(y) \geq u_i(x)\) if \(y \geq x\). For the initial endowment of some trader \(i\), assume \(w_{ij} > 0\) for at least one \(j\). At a given \(\pi \in \mathbb{R}^N_+\), trader \(i\) will sell their endowment in ex-

---

\(^1\)Following the exposition of Herbert Scarf\(^20\) as I do so.

\(^2\)Scarf\(^19\) and Dantzig\(^8\) are notable exceptions, along with Smale\(^23\)\(^22\), who developed a global Newton’s method for the computation of equilibrium prices by counting the elements of the Jacobian of the excess demand correspondences. See also Velupillai\(^26\) for a collection of essays on the topic.
change for some bundle of goods, \( x_i = (x_{i1}, \ldots, x_{in}) \in \mathbb{R}^N_+ \), which maximises \( u_i(x) \) subject to their budget constraint: \( \pi \cdot x \leq \pi \cdot w_i \). An equilibrium is a vector of prices \( \pi = (\pi_1, \ldots, \pi_n) \in \mathbb{R}^N_+ \) at which, for each trader \( i \), there exists a bundle \( \bar{x} = (\bar{x}_{1i}, \ldots, \bar{x}_{ni}) \in \mathbb{R}^N_+ \) of goods such that two conditions hold:

1. \( \forall i \), the vector \( \bar{x}_i \) maximises \( u_i(x) \), subject to the constraints \( \pi \cdot x \leq \pi \cdot w_i \), and \( x \in \mathbb{R}^N_+ \).
2. For each good, \( j \), \( \sum_i \bar{x}_{ij} \leq \sum_i w_{ij} \).

For any price vector \( \pi \), a vector \( x_i(\pi) \) which maximises \( u_i(x) \) subject to the budget constraint \( \pi \cdot x \leq \pi \cdot w_i \) and \( x \in \mathbb{R}^N_+ \), is called a demand of trader \( i \) at prices \( \pi \). When there exists a demand vector, we call the locus of points generated the demand function. All of the above reasoning is true, \textit{pari passu}, for quasi-concave utility function formulations of the general equilibrium problem. It should also be noted that the demand function is, strictly speaking, a correspondence rather than a single valued function\(^{16}\).

Call the vector \( z_i(\pi) = x(i) - w_i \) the excess demand of individual \( i \) or, as McCauley\(^{14}\) puts it, let us measure our dissatisfaction. When individuals trade, let \( X^k(\pi) = \sum_i x_{ik}(\pi) \) denote the market demand for good \( k \) at prices \( \pi \). Market excess demand is therefore \( Z^k(\pi) = X^k(\pi) - \sum_i w_i k \). The vector \( X(\pi) = (X^1(\pi), \ldots, X^N(\pi)) \) and \( Z(\pi) = (Z^1(\pi), \ldots, Z^N(\pi)) \) are called market demand and excess demand correspondences, respectively. Including the assumption of homogeneity of demand correspondences (implying \( \forall \lambda > 0, Z(\pi) = \lambda Z(\pi) \)), the assumptions we place on the utility functions guarantee \( \pi \cdot x_i(w) = \pi \cdot w_i \) at prices \( \pi \). This satisfies Walras’ law such that at prices \( \pi, \pi \cdot Z(\pi) = 0 \).

The equilibrium is therefore the vector of prices \( \pi = (\pi_1, \ldots, \pi_N) \in \mathbb{R}^N_+ \) such that \( Z^j(\pi) \leq 0 \) for each \( j \). Foley\(^{9}\) discusses the possibility of the realised price vector emerging from a probability distribution, lending a statistical mechanics interpretation to the determination of prices, a subject which has recently been reinvigorated through the nascent econophysics tradition\(^{29}\).

Samuelson\(^{18}\) and others formalised the process of convergence from given or initial utility functions and endowments to the acquisition of equilibrium coordinates via the specification of the following differential equation:

\[
\frac{d\pi_k}{dt} = G_k(Z_k(\pi)), \quad \forall \quad k = (1, 2, \ldots, N).
\]

For equation 2.1, assumptions on the nature of \( G_k(\cdot) \) vary substantially\(^{19} \text{20}\), but the function must be continuous, differentiable everywhere, and sign preserving at least\(^1\) to pick out the appropriate excess demand function \( Z_k(\cdot) \) for some \( k \). The major question, of course, is whether the solution, starting from initial disequilibrium conditions, will emerge from the trading process. We have returned, circuitously, to the optimisation problems mentioned in section 1.

\(^1\)Velupillai has of course much to say about the assumption of completeness underlying the Hahn Banach theorem.
Velupillai makes much of the algorithmic economist’s ability to strengthen a hypothesis or weaken a conclusion in order to achieve a numerically meaningful result, and his technique for doing so is both powerful and subtle. Taking his example, we see the standard intermediate value theorem (IVT):

**Theorem 2.1 (Intermediate Value Theorem).** Let \( f : \mathbb{R} \rightarrow \mathbb{R} \) be a continuous function from the closed interval \([a, b]\) to \(\mathbb{R}\) and let \( f(a) < 0 \) and \( f(b) > 0 \). Then \( \exists x \in \mathbb{R} \) and \( a < x < b \) such that \( f(x) = 0 \).

In the hands of an algorithmic economist, this formulation of the IVT has no constructive construal, and hence either the conclusion must be weakened or the hypothesis strengthened to square the theorem with a natural, numerical, and algorithmic interpretation. Following Velupillai, we must delimit the range of the theorem, introducing approximants in the form of \( \epsilon \) and ask ourselves what data types this theorem will admit in order to prove that the reals are algebraically closed. Velupillai shows us the formulation of what I shall call the Constructive Intermediate Value Theorem (CIVT) to be:

**Theorem 2.2 (Constructive Intermediate Value Theorem).** Let \( f : \mathbb{R} \rightarrow \mathbb{R} \) be a continuous function, from \([a, b] \subset \mathbb{R}\) to \(\mathbb{R}\), and let \( f(a) < 0 \), \( f(b) > 0 \); then \( \forall \epsilon > 0, \exists x \in \mathbb{R} \) such that \( a < x < b \) subject to \( |f(x)| < \epsilon \).

We see how a strengthening of the hypothesis (by assuming \( f(x) \) to be a polynomial) and weakening the conclusion by allowing a precisely numerical interpretation of an approximation to the \( x \) being sought creates a numerically meaningful and algorithmically defensible CIVT.

I have written that Velupillai provides a blueprint for algorithmic economics in his paper. Here is why: we can use the technique shown above in moving from theorem 2.1 to theorem 2.2 to judiciously build an algorithmic economics out of mainstream economics, first by asking the standard questions economists ask using recursion theory (this is computable economics), and secondly by allowing the use of precisely determined approximations to modify the standard approaches. This is algorithmic economics.

Returning to the general equilibrium problem stated above in section 2.1, let us state the general equilibrium problem again: we must find a price vector \( \pi \) such that \( Z'(\pi) \leq 0 \) for each \( j \). We must recognise first that, however they are actually discovered, most equilibrium price vectors will not be exact equilibria. We can thus search for prices which satisfy the following search problem, itself a variant of Scarf’s method.

**Remark 2.1 (Search Problem).** A bundle \( x_i \in \mathbb{R}^N_+ \) is an approximate demand if, for \( \epsilon \geq 1 \), each trader \( i \), at prices \( \pi \), \( u_i(x_1) \geq \frac{1}{\epsilon} u^* \), and \( \pi \cdot x_i \leq \epsilon \pi \cdot u_i \), where \( u^* = \max\{u_i(x)|x \in \mathbb{R}^N_+, \pi \cdot x \leq \pi \cdot i\} \).

A price vector in this sense will be an \( \epsilon \)-approximant to the ‘true’ equilibrium if there exist bundles \( x_i \) for which the following conditions hold:
(1) for each and every trader \( i \), \( x_i \) is the demand of trader \( i \) at prices \( \pi \), and;
(2) \( \sum_i x_{ij} \leq \epsilon \sum_i w_{ij}, \forall (i, j) \).

At every stage of the computation, we are aware of \( \epsilon \). It, in effect, constitutes our 'stopping rule', where we decide how close to the equilibrium we would like to be. Velupillai’s article offers hints on how to develop this theory further, but it remains an open question and a fertile field for research.

So we weaken the hypothesis, and allow a precisely determined approximation to strengthen the conclusion. There is an intellectual pedigree for such work: Brouwer himself ‘recast’ an intuitionistically valid version of his famous fixed-point theorem in this form:\(^4\)

**Theorem 2.3 (Intuitionist Fixed Point Theorem).** For each continuous self map of the closed unit disc \( D \in \mathbb{R}^2 \), and each \( \epsilon > 0 \), there exist some \( x \in D \) such that \( ||f(x) - x|| < \epsilon \).

Brouwer\(^6\) termed these \( \epsilon \) injections “geodetic distances”, so sticking with his original terminology, I will use the rest of this paper to argue for a taxonomy of geodetic distances in algorithmic economics. Velupillai does not explore this topic, but extolls the virtues of including arbitrarily precise approximants. These geodetic distances already abound in economics, as Mcullough and his collaborators discovered in their overview of econometric and computable general equilibrium software: to actually find fixed points which are the roots of large polynomials being estimated for policy purposes, these \( \epsilon \) approximants are required. Bringing the role these approximations play into the forefront of discussion about economic modeling would go a long way towards making economics an algorithmic economics.

Why would this be so? First, having natural and algorithmic interpretations of theorems in economics which allow such approximations allows us to discuss the complexity of these algorithms, and their feasibility in real world processes. Second, having a taxonomy of these \( \epsilon \)s would induce a more truthful and scientific reporting process to the estimation methods in economics currently. Third, once the foundational algorithms of the discipline are built and their upper and lower bounds tested, refinements, advancements, and new sub branches may be added. We would have a truly algorithmic economics, because that is where researchers would start.

3. Conclusion: Computable, constructive, and Intuitionist distinctions

‘When I use a word,’ Humpty Dumpty said, in a rather scornful tone, ‘it means just what I choose it to mean, neither more nor less.’

‘The question is,’ said Alice, ‘whether you can make words mean so many

\(^4\) This formulation of Brouwer’s original theorem is due to Bridges\(^3\).
different things.'

‘The question is,’ said Humpty Dumpty, ‘which is to be master - thats all.’

The quote from Lewis Carroll’s *Through the Looking Glass* gives a whimsical opening to this section. As there are varieties of approaches in economics, so there are varieties of approaches to computability in economics and, more properly, mathematics.

In modern mathematical economics, an object exists if its existence is non-contradictory. The majority of theorems proved in mathematical economics have this non-contradictory core, and so only show existence by this route. The key difference between mainstream mathematical economics and algorithmic economics is this definition of existence. A mathematical object cannot be said to exist if there does not exist an algorithm which, at least in principle, could construct that object.

In constructive mathematics, there are several schools of thought on methods to determine whether an object exists. Take the notion of an algorithm, which we can define as a step by step procedure which can in principle be complete in a finite period of time.

In the type of constructive analysis developed by Erret Bishop, and the intuitionistic mathematics developed by L. E. J. Brouwer in his later years, the algorithm is a primitive object existing outside of any formal language. Russian constructivism, as Bridges and Richman aver, only acknowledges the existence of an algorithm within a given programming language, thus an algorithm is a description of the motion of symbols within a language. The type of constructivism one wishes to use matters, as Velupillai describes. The lessons of each school, broadly, accord with algorithmic economics and it’s proper subset, computable economics. The key difference between these approaches and algorithmic economics is the appeal to the fundamental numerical character of mathematics, echoing Schumpeter’s quote above, and Bishop’s statement that

\[\text{[t]he primary concern of mathematics is number, and this means the positive integers \ldots} \text{Everything attaches itself to number \ldots} \text{Every mathematical statement ultimately expresses the fact that if we perform certain computations within the set of positive integers, we shall obtain certain results.}\]

I believe the fundamental character of economics to be numerical, which is why I believe an algorithmic economics will be a fruitful research program. The question is clearly: which is to be master?

4. Conclusion

Velupillai shows us a method by which one can convert many of the theorems in mathematical economics to give them algorithmic interpretation. First, begin

\[\text{1Encompassing, naturally, computable and constructive economics}^{25}.\]
with the idea that one’s assumptions and axioms constitute the inputs to be used within the algorithm/theorem. Then, visualise the theorem itself as the output of a constructive proof process (an algorithm). Consider the proof as the discrete series of steps to get one from the inputs to the outputs. Where a stumbling block occurs in the construction of the proof, it is most likely the use of the law of the excluded middle which is interfering with the proof’s completion. One can’t know what one can’t show.

This paper has shown that the difference between the mainstream and algorithmic approaches to mathematical economics is foundational, because each approach has a different process for assessing what it means for an object to exist. The algorithmic route, although more arduous, is more fruitful, because once one has a procedure by which one can reach a desired state—the theorem, in principle, one can test that assertion in the real world without recourse to ad hoc truncations, approximations and numerical analysis. One simply pumps real world numbers into the algorithm, and observes the results. The results of the theorem have instant numerical meaning, and can be assessed accordingly. This blueprint of Velupillai’s for algorithmic economics has instant and lasting value for that reason.

References


Which states for some proposition \( P \): \( P \lor \neg P \) implies existence of \( P \).
28. K. V. Velupillai. Taming the incomputable, reconstructing the nonconstructive and deciding the undecidable in mathematical economics. New Mathematics and Natural Computation, forthcoming.