Formalisation of Economics through Type Theory

Iván Rendo Barreiro, *M1 in Econometrics and Statistics - Decision Mathematics*

***Abstract*—The main objectives of the thesis will be: first, to discuss whether there is room for a formalisation of economic theory through type theory. Second, -if possible- to outline how such formalisation could be carried out and, finally - if points 1 and 2 are accomplished-, to define some basic concepts and proof some elementary results of any branch of the economic theory (to be chosen) in a proof assistant[1].**

I. Why? What is achieved with this?

In short: proof checking and automation[2], which are highly interconnected.

Through formal verification, a computer checks that all the results used in a proof are derived from each other and from existing definitions and it is able to find logical gaps or errors, as it is the case here.

Taking all these verified proofs as a database, one can believe that it is possible to teach a computer to demonstrate (for now, they only have certain *dumb* mechanisms called tactics). However, for the QED manifesto, this is not an objective *per se*.

In economics...

I do not know at this moment. I cannot imagine the power (or lack of power) of this approach in economics. It is an interesting point to be treated. As far as I know, some theorists (at least) use this kind of software as a proof checker.

Nevertheless, this type of approach does appear quite a lot in social choice theory. For example, here.

II. Which methodology is used?

In practice, this formalisation is carried out mainly through writing code in a proof assistant. There are several available and well implemented. Some of the most famous are Isabelle, INRIA's[3] Coq and Microsoft Research's Lean. At this time, I am more experienced with Lean, although -as a current member of the French university system- it might be a good idea to work with Coq, created by Thierry Coquand. However, this choice does not seem to be too important.

III. How is a proof implemented in Lean?

It is difficult to explain in words. Here is a video where Scott Morrison proves the infinitude of primes. Just out of curiosity, this is what the proof of the statement *'two natural numbers are equal if and only if their successors are equal'* looks like in Lean[4]:

```
theorem test_prop (a b : mynat) : succ a =
succ b <-> a = b :=

begin
split, exact succ_inj, intro g,
rw g, refl,
end
```

IV. What lies behind?

Dependent Type Theory. From the first lines of the Lean documentation *'Type theory' gets its name from the fact that every expression has an associated type* and the 'dependent' surname indicates that each type may depend on a parameter. For example, we can define a term x which type is 'Natural' if we want to state that $x \in \mathbb{N}$. Moreover, it is possible to define the type 'Vector α n', the type of vectors of type α (first parameter) of lenght n (second parameter).

It seems very intuitive up to here, but there is much more behind it and -at the moment - I am far from being an expert in type theory. This is presumably one of the main issues to be addressed during the preparation of the thesis.

Up to this point, type theory and set theory seem very similar. This entry from the Stanford Encyclopedia of Philosophy compares both.

Disclaimer and Conclusion

I am not an expert on any of these issues at all. However, I would love to learn more about and the intersection between economics, mathematics and computer science. Especially when it comes to formalisation. For the previous reason, I think this is a risky bet (as it may be too much for me). However, I honestly believe that this -the M1 thesis- is a suitable enviroment to try something like this, as the potential benefits are quite high.

Why do I think this type theory in economics makes sense? Because of the K. Buzzards's Xena Project. This project aims to formalise all the mathematics taught in Imperial's undergraduate programme. Why not economics?

Why am I worried that this might not make sense? I have not found anything that had much to do with this approach in economics. This is what I find surprising. Except these slides on 'Economic Equilibria in Type Theory' (both games and Walrasian).

References

[1] Smith, S. F. (1988). *Partial objects in type theory.* Cornell University.

Written January 12, 2021.

[1] My first idea was to prove Gale-Nikaido-Debreu's theorem of the existence of competitive equilibrium, which was the first result I saw formally proved in economics. However, *a priori*, any should be valid.

[2] There are many others. E.g. it is used in the verification of some digital systems or cryptographic protocols.

[3] *Institut National de Recherche en Informatique et en Automatique.*

[4] For more examples, here is the documentation of the user-maintained library 'mathlib' on Extrema of Smooth Functions.