Economic Equilibria in Type Theory

Cezar Ionescu

Domain-Specific Languages

Examples of domain-specific languages.

Domain-Specific Languages

Examples of domain-specific programming languages.

Domain-Specific Languages

Examples of domain-specific languages.

Domain-Specific Languages

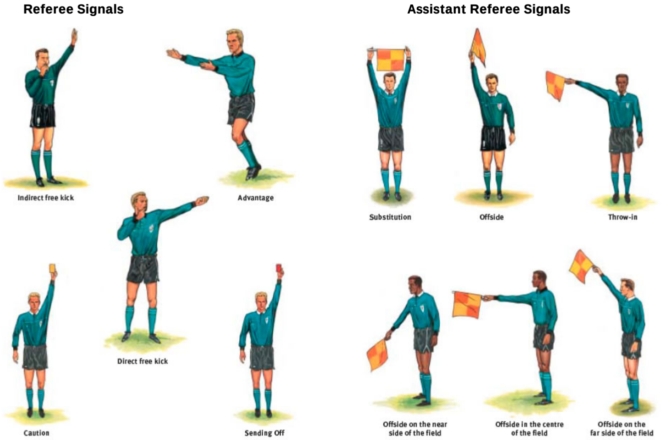
Examples of domain-specific gesturing languages.

Domain-Specific Languages: Baroque Gestures

Hesitation

Curiosity

Domain-Specific Languages: Referee Signals



Domain-Specific Languages

Examples of domain-specific gesturing languages.

Domain-Specific Languages

Examples of domain-specific languages.

Domain-Specific Languages

Examples of domain-specific mathematical languages.

Domain-Specific Languages in Mathematics

Each branch of mathematics has its own set of concepts and methods: its own language.
Bourbaki: but every such language has set-theoretical semantics.

Domain-Specific Languages in Mathematics

Example: plane euclidian geometry.

- the primitive terms, points, straight lines, planes, are interepreted in terms of sets
- the primitive operations (intersection of straight lines, being on a straight line, etc.) are interpreted in terms of operations on these sets
- the only constraint on the interpretation is that the euclidian axioms are translated as true sentences of set theory.

Domain-Specific Languages in Mathematics

Other examples: linear algebra (vector spaces and linear transformations), topology (topological spaces and continuous functions), group theory (groups and morphisms), etc.

Category theory proved useful in organizing some of these languages (and also in organizing some of the programming DSLs).

Set theory appears as a sort of machine-language of mathematics.

Domain-Specific Languages in Mathematics

Another example: economics.

Concepts such as welfare, supply and demand, equilibrium, market, etc.

Question: how can we use the mathematical DSL of economics to obtain a DSL for programming economic models?

Enter the Curry-Howard Isomorphism

The *Curry-Howard isomorphism*:

- a set of theorems (by Curry, Howard, de Bruijn, and many others) relating certain formal systems for intuitionistic logic with corresponding lambda calculi
- a justification for an operational reading of not fully formalized statements of (constructive) mathematics.

The Curry-Howard Isomorphism and the BHK Interpretation

Example:

Mathematical statement: there exist irrational numbers x and y such that x^y is rational.

Operational reading: find (construct, build a machine that produces) irrational numbers x and y such that x^y is rational.

The Curry-Howard Isomorphism and the BHK Interpretation

Example:

Mathematical statement: there is no largest prime number.

Operational reading: Given a machine which produces the largest prime number, construct a machine that produces an element of the empty set.

The Curry-Howard Isomorphism and the BHK Interpretation

Example:

Mathematical statement: If an allocation-price pair $(\mathbf{x}, \mathbf{p})$ is a Walrasian equilibrium, then $\mathbf{x}$ is Pareto efficient.

Operational reading: Given a machine that produces Walrasian equilibria, construct a machine that produces Pareto efficient allocations.

Walrasian equilibrium

Definition of Walrasian equilibrium. An allocation-price pair $(\mathbf{x}, \mathbf{p})$ is a **Walrasian equilibrium** if (1) the allocation is feasible, and (2) each agent is making an optimal choice from its budget set. In equations:

1. $\sum_{i=1}^{n} \mathbf{x}_i = \sum_{i=1}^{n} \boldsymbol{\omega}_i$
2. If $\mathbf{x'}_i$ is preferred by agent i to $\mathbf{x}_i$, then $\mathbf{px'}_i > \mathbf{p}\boldsymbol{\omega}_i$.

Varian, p. 325

Walrasian equilibrium

An allocation is represented by a vector of positive real numbers:

$\mathbf{x}_i \in \mathbb{R}^m_{\geq 0}$

where m is the number of *goods*. The real number $\mathbf{x}_{ij}$ represents the quantity of good j that agent i has.

Walrasian equilibrium

Similarly, prices are represented by a vector of strictly positive real numbers:

$$\mathbf{p} \in \mathbb{R}^m_{>0}$$

where m is the number of *goods*. The real number $\mathbf{p}_j$ represents the price of good j.

Walrasian equilibrium

There are *initial endowments* for the agents, represented by positive real numbers:

$\boldsymbol{\omega}_i \in \mathbb{R}^m_{\geq 0}$

$\boldsymbol{\omega}_{ij}$ represents the initial quantity of good j that agent i has.

Walrasian equilibrium

The feasibility condition for a (re)allocation $\mathbf{x}$ means that the quantity of goods is conserved:

$$\sum_{i=1}^{n} \mathbf{x}_i = \sum_{i=1}^{n} \boldsymbol{\omega}_i$$

Walrasian equilibrium

The agents have preferences on their own bundles of goods (there are as many preferences as there are agents).

The optimality condition means that any bundle of an allocation x' which the agent would prefer to the optimal bundle x_i would cost more than the agent can spend:

$$\mathbf{p}\mathbf{x'}_i > \mathbf{p}\boldsymbol{\omega}_i.$$

(there is again a conservation implicit in the optimality).

Walrasian equilibrium

Now let's see what this looks like in Agda . . .

Pareto efficiency

Definitions of Pareto efficiency. A feasible allocation $\mathbf{x}$ is a **weakly Pareto efficient** allocation if there is no feasible allocation $\mathbf{x}'$ such that all agents strictly prefer $\mathbf{x}'$ to $\mathbf{x}$. A feasible allocation $\mathbf{x}$ is a **strongly Pareto efficient** allocation if there is no feasible allocation $\mathbf{x}'$ such that all agents weakly prefer $\mathbf{x}'$ to $\mathbf{x}$, and some agent strictly prefers $\mathbf{x}'$ to $\mathbf{x}$.

Varian, p. 323

Weak Pareto efficiency

The notion implied in the first welfare theorem is that of *weak* Pareto efficiency.

Pareto efficiency is more general than Walrasian equilibrium: there is no constraint on what "feasible" means.

Similarly to Walrasian equilibrium, each agent is assumed to have its own preference, but this time the preferences are on *allocations*, not just on individual bundles.

Weak Pareto efficiency

Similarly to Walrasian equilibrium, we have a feasiblity condition and an optimality one:

If *all* agents prefer $\mathbf{x}'$ to $\mathbf{x}$, then $\mathbf{x}'$ is not feasible.

Let's see how that looks in Agda . . .

First welfare theorem

In order to state the first welfare theorem precisely, we need to specify what feasibility condition and what global preferences we'll use in order to "fill in" the context of Pareto efficiency.

For feasibility, we use the same conservation condition.

For preferences, we need to translate the "local" preferences of the agents to "global" ones:

agent i prefers allocation $\mathbf{x}$ to allocation $\mathbf{x'}$ if it prefers its bundle in $\mathbf{x}$ over that in $\mathbf{x'}$.

Preferences

Preferences are supposed to be total preorders (reflexive and transitive).
The relation obtained by extending the "local" preferences to allocations is a total preorder.

But we have to prove this to Agda . . .

First welfare theorem

Finally, we can now state the first welfare theorem in Agda . . .

Informal proof of the first welfare theorem

If $\mathbf{x'}$ is feasible, then $\sum_{i=1}^{n} \mathbf{x'}_i = \sum_{i=1}^{n} \boldsymbol{\omega}_i$.

Multiplying both sums by $\mathbf{p}$, we have that allocation $\mathbf{x'}$ costs just as much as $\boldsymbol{\omega}$.

However, since $\mathbf{x}$ is optimal, and since we are given that every agent strictly prefers $\mathbf{x'}$ to $\mathbf{x}$, we have that each individual bundle in $\mathbf{x'}$ costs more that the respective bundle in $\boldsymbol{\omega}$. Therefore, the allocation $\mathbf{x'}$ costs strictly more that the initial endowment.

Contradiction.

Formal proof of the first welfare theorem

We can implement the informal argument almost literally in Agda

...

Conclusions

The formal proof of the first welfare theorem gives us a program to construct certain Pareto efficient allocations from Walrasian equilibria.

The construction is automatic, all parts are guaranteed to be correct (e.g. the preferences are total preorders).

This is has been achieved in a systematic way, by giving an operational reading to established mathematical concepts via the Curry-Howard isomorphism.

We are currently playing the same game with more interesting equilibria of the Nash "variety": simple, mixed, correlated, local (Flondor). In particular, we can test codes that claim to compute correlated equilibria (Gintis) by testing to see how well they do with simple Nash.

Conclusions

An unsatisfactory aspect so far is that we are alway operating in a conditional mode, e.g., if you have a way of computing minima, then we can construct a way of computing equilibria

One reason for this is that our operational reading is often "blocked" by non-constructive proofs. This forces a difference between the classical part, which can only be approximated, and the constructive part, which can be computed.

Ultimately, we'd like to replace set theory with type theory as machine language: then we'd have a unitary development of the math and the programming.

But we can go a long way with conditional correctness (see McBride 2011).