SEMANTICS FOR THE λ-CALCULUS

ARNOUD VAN DER LEER
DELFT UNIVERSITY OF TECHNOLOGY
ARNOUDVANDERLEER@GMAIL.COM

1. Intro

Hi, it is so good to see you all here. For those who don't know me, I am Arnoud van der Leer, master's student at TU Delft, and am currently working on my master's thesis, supervised by Benedikt and Kobe.

Today, I am going to talk about the paper 'Classical lambda calculus in modern dress'. This is a paper from 2013, written by Cambridge professor Martin Hyland. In this paper, he talks about models for the λ-calculus, and proves three 'big' theorems about them. I will also talk a little bit about my thesis, which revolves around this paper. My job is to 'annotate' this paper, so to speak.

Contents

So here is the outline of my talk. We are currently in the introduction. To prove things about the λ-calculus and its models, we first need establish the way in which we will formally talk about the λ-calculus. We will also talk a bit about denotational semantics. Then we will have a look at the three main theorems. The first main theorem is Scotts representation theorem, which shows that every model of the λ-calculus can be obtained from some category. Hyland calls the second main theorem the fundamental theorem of the λ-calculus. This is a somewhat stronger result, which says that there is an equivalence between models and denotations for the λ-calculus. For me, the last main theorem seems to be the most cryptic one, to me. It states that if we have a model for the λ-calculus, then create its 'category of retracts', then this category is 'relatively cartesian closed'. It basically says: if we have a model for the λ-calculus, we can create a category in which we can do some form of dependent type theory. Lastly, I will talk a bit about my contribution to this, so annotating the paper, and especially mechanizing it.

2. Talking about the λ-calculus

So, first of all, in this talk, I will talk almost exclusively about the **untyped** λ-calculus. The untyped λ-calculus essentially is about collections that consist of only functions. With the objects of these collections, (so functions), we can do a couple of things. First of all, we can create new objects, which are just the 'variables'. Then, given two of these objects, we can apply one to the other. Lastly, given a function with $n + 1$ free variables, we can create a function with n free variables by abstracting.

Again, note that everything here is a function.

We can impose two types of equalities on these terms: β-equality and η-equality. One is about what happens when you first abstract and then apply. The other is about what happens if you first apply to a variable and then abstract again.

So a model for the λ-calculus is something that exhibits this structure. Something thas has variables, application and abstraction, and maybe β and/or η-equality.

One such collection is the *pure* λ-calculus. One can view this as an inductive type, given by these three constructors. We can also create the pure λ-calculus with β-equality, which arises from this inductive type via a quotient.

2.1. Models. So, the first stepping stone in talking formally about models are algebraic theories. These are structures with variables and some substitution operation.

One example of this is the Λ-calculus.

Another example is a polynomial ring. It has variables, you can add and multiply polynomials together. And if you have two polynomials, you can substitute one in the place of a variable in the other to get a new polynomial.

The formal definition of an algebraic theory is

2.2. Semantics.

3. The main theorems

3.1. Scott's representation theorem.

3.2. The fundamental theorem of the λ-calculus.

3.3. The category of retracts.

4. My contribution

4.1. Annotating the paper.

4.2. Mechanization.

5. Conclusion