Completeness for identity free Kleene Lattices

Amina Doumane

February 11, 2019

1 Motivation

- I will start by giving some motivations for this work, although I dont think it is really necesary for this audience.
- The motivtion for this work comes from the task of verification of imperative programs. These programs are lists of instructions. A pleasant way to think about each of these atomic instructions is a relations which transforms the memory states. For instance, the first command can be seen as a relation calle it a, etc... And the whole program can be seen as the composition of these relations.
- Now, suppose that we have two imperative programs, and suppose that we want to check if they have the same behaviour. For insatnce let us take the program $x \leftarrow 1; ((y \leftarrow x) + (y \leftarrow 0))$ and $(x \leftarrow 1; y \leftarrow x) + (x \leftarrow 1; y \leftarrow 0)$.
- To check that two programs have the same behaviour, one possible way to proceed is to abstract these instructions into relations (here a, b and c) and check that the obtained expressions are equal in the relational model. This means that no matter how we chose to interpret a, b, c as relations, the equation still hold. If this is the case, we can be sure that the two programs have the same behaviour.
- In this example, I used relations composion and union to simulate programs composition and non-deterministic choice. But depending on the class of programs under consideration, we may need other operators, for instance transitive closure to simulate while loop, (et les autres?) In this presentation I will focus on two classes of operators: the first one is the set of KA operators. If we look at the expressions generated by these operations we get regular expressions. This class of operators is very standard, and will serve as a reference point. The second class of operators are the KLm operators. It contains the operators of KA except the identity(1) together with the intersection operator. And I introduce it because my completness result will be about this fragment. Here is an example of a valid law in KLm. Note that in the first law I considered the equality relation, while I considered inclusion in the second one. Both are interesting.