Program Embeddings

1 Introduction

Introduction text

Language Model

Here we describe a simplistic programming language model commonly seen in the programming language community. This model is sometimes called IMP, for **IMP**erative.

IMP

$$
\begin{array}{lll}
\bigoplus & \texttt{:=} & f \in \{+, *\} \\
a & \texttt{:=} & n \in \mathbb{R} \mid x \in \texttt{Var} \mid \bigoplus(a_1, \ldots, a_n) \\
b & \texttt{:=} & \top \mid \bot \mid b_1 \wedge b_2 \mid \neg b \\
c & \texttt{:=} & \texttt{skip} \\
 & \mid & x \leftarrow a \\
 & \mid & \texttt{if}\ b\ \texttt{then}\ c_t\ \texttt{else}\ c_f \\
 & \mid & \texttt{while}\ b\ \{c\} \\
 & \mid & c_1 \texttt{;} c_2
\end{array}
$$

Figure 1: IMP grammar

The semantics are as follows:

2 Restriction 1

We now consider the following restriction of IMP:

$$\frac{\Gamma[x] = n}{(x, \Gamma) \hookrightarrow n}$$

$$\frac{\forall i \ : (a_i, \Gamma) \hookrightarrow n_i}{(\bigoplus (a_1, \ldots, a_n), \Gamma) \hookrightarrow \bigoplus (n_1, \ldots, n_n)}$$

$$\frac{r \in \mathbb{R}}{(\bigoplus (n_1, \ldots, n_n), \Gamma) \hookrightarrow r}$$

Figure 2: IMP arithmetics

Figure 3: Restriction 1 IMP grammar

3 Kleene Algebra

3.1 Test

Note: there's an embedding of KA into LA through the automata interpretation, but how do we preserve the semantics of the conditionals under which the splitting happens?