Term Algebras

assume $(\{f_1,\dots,f_n\}, ar)$ $\quad ar: \{f_1,\dots,f_n\} \to \omega$

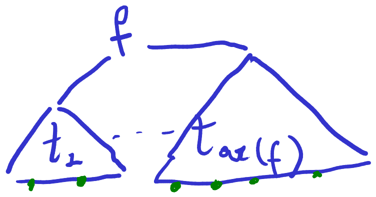
$\mathcal{V} \cap \{f_1,\dots,f_n\} = \emptyset$

Term Algebra The term algebra on a signature Σ and a countable set $\mathcal{V}$ of variables is the smallest set $Term_{\Sigma,\mathcal{V}}$ s.t.

- $\mathcal{V} \subseteq Term_{\Sigma,\mathcal{V}}$
- $\forall f \in \Sigma,\ t_1,\dots,t_{ar(f)} \in Term_{\Sigma,\mathcal{V}} :\ f(t_1,\dots,t_{ar(f)}) \in Term_{\Sigma,\mathcal{V}}$

$T_\Sigma = Term_{\Sigma,\emptyset} \subseteq Term_{\Sigma,\mathcal{V}}$ is the set of closed terms

Exercise 7
Explain why in the above definition it is essential to require that $Term_{\Sigma,\mathcal{V}}$ is the smallest set



- are either variables or "constants" (i.e $c \in \Sigma$ s.t $ar(c) = 0$)

Exercise 8
Give the term algebra for regular expressions

Transition System Specifications

Aceto, Fokkink, Verhof: Structural Operational Semantics

"The first systematic study of TSSs may be found in [208], while the first study of TSSs with negative premises appeared in [57]." (Aceto et al.)

[208] R. d. Simone, Calculabilité et Expressivité dans l'Algèbre de Processus Parallèles Meije, thèse de 3 e cycle, Univ. Paris 7, 1984.

[57] B. Bloom, S. Istrail, and A. Meyer, Bisimulation can't be traced: preliminary report in Conference Record 15th ACM Symposium on Principles of Programming Languages, San Diego, California, 1988, pp. 229–239. Preliminary version of Bisimulation can't be traced, J. Assoc. Comput. Mach., 42 (1995), pp. 232–268.

Fix a term algebra $Term_{\Sigma,V}$ and a set of labels A

A TSS with labels A is a set of (inference) rules $\dfrac{H}{\alpha}$

finite set of literals

positive literals

LITERALS

positive

$\underbrace{t \xrightarrow{a} t' \quad \text{or} \quad t \in X}$

negative

$\underbrace{t \stackrel{a}{\not\rightarrow} \quad \text{or} \quad t \notin X}$

where $a \in A$, $t \in Term_{\Sigma,V}$, $X \subseteq Term_{\Sigma,V}$

Operational semantics of regular expressions

A TSS

(Act) $\dfrac{a \in A}{a \xrightarrow{a} 1}$

(tic) $\dfrac{}{1 \xrightarrow{\varepsilon} 1}$

(Seq$_1$) $\dfrac{x \xrightarrow{a} x' \quad x' \neq 1}{x \cdot y \xrightarrow{a} x' \cdot y}$

(Seq$_1$) $\dfrac{x \xrightarrow{a} 1}{x \cdot y \xrightarrow{a} y}$

(Cho$_1$) $\dfrac{x \xrightarrow{a} x' \quad x' \neq 1}{x + y \xrightarrow{a} x'}$

(Cho$_2$) $\dfrac{x \xrightarrow{a} 1}{x + y \xrightarrow{a} 1}$

(Cho$_3$) $\dfrac{y \xrightarrow{a} y' \quad y' \neq 1}{x + y \xrightarrow{a} y'}$

(Cho$_4$) $\dfrac{y \xrightarrow{a} 1}{x + y \xrightarrow{a} 1}$

(Star$_1$) $\dfrac{}{x^* \xrightarrow{\varepsilon} 1}$

(Star$_2$) $\dfrac{x \xrightarrow{a} x'}{x^* \xrightarrow{a} x' \cdot x^*}$

Note that
- x & y range over the set of reg exp
- these rules form a TSS
- each operator has a set of rules (including 0, which has ∅!)

Basic Process Algebras with $a \in A \cup \{\varepsilon\}$

Exercise 9
Simplify the TSS above (Hint: Think about the rules for choice)

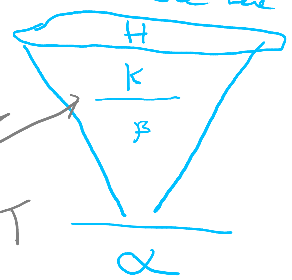
LTSs as proofs of TSSs

A proof in a TSS T of a closed transition rule H/α is an upwardly branching tree without infinite branches, whose

are the leaves

- nodes are labelled by literals
- the root is labelled by α, and
- if K is the set of labels of the nodes directly above a node with label β, then
  1. either K = ∅ and β ∈ H,
  2. or K/β is a closed substitution instance of a transition rule in T.

closed subst instance of a rule in T



If a proof of H/α from T exists, then H/α is provable from T , notation T |-- H/α.

Exercise 10
Formally define closed-term substitutions and their application to terms of a term algebra.