Term Rewriting Systems

Friederike Jolk

Universität Bremen
WS 2005/06
Seminar “Transformationen”

9.12.2005
Basic Notions
  Signature and Terms
  Subterms and Transformed Terms

Reductions
  Identities and Reduction Relations
  Substitutions

Term Rewriting Systems
  Rewrite Rules and Term Rewriting Systems
  Properties
Basic Notions

- Signature $\Sigma$ – set of function symbols
- Arity $n$, associated to an $f \in \Sigma$
- $X$ – set of variables
- Term $T(\Sigma, X)$ – set of terms over $\Sigma$ and $X$

```
data Term = V Char | T Char [Term]
```

- $Var(t)$ – set of variables occurring in Term $t$
  - obviously $Var(t) \subseteq X$
Basic Notions (cont’d)

- Positions $\text{Pos}(t)$ – set of positions of term $t$

```haskell
pos :: Term -> [String]
pos V x = []
pos T f ts = [] ++ [(show i) ++ p | i <- [1..n],
    t <- ts, p <- pos t, index t ts == i-1 ]
where n = length ts
    index t ts = findIndex (==t) ts
```

- Subterm $t \mid_p$ – subterm at position $p$
- Transformed Term $s[t]_p$ – term obtained from $s$ by replacing the subterm $s \mid_p$ by $t$
Basic Notions
   Signature and Terms
   Subterms and Transformed Terms

Reductions
   Identities and Reduction Relations
   Substitutions

Term Rewriting Systems
   Rewrite Rules and Term Rewriting Systems
   Properties
Reduction Relations

- **Identity** $(t_1, t_2) \in T(\Sigma, X) \times T(\Sigma, X)$
  - usually denoted by $t_1 \approx t_2$
  - may be read bi-directional

- **Relation** $\rightarrow: A \times B$
  - $\rightarrow (a, b)$ usually denoted by $a \rightarrow b$ for $a \in A, b \in B$
  - considered as directed

- **Reduction Relation** $\rightarrow_E \subseteq T(\Sigma, X) \times T(\Sigma, X)$
  - stepwise “decrease” of something, e.g. number of variables
Substitution

- of a Variable: \( \sigma : X \rightarrow T(\Sigma, X) \)

\[
\text{substVar :: Subst} \rightarrow \text{Char} \rightarrow \text{Term} \\
\text{substVar} [] x = V x \\
\text{substVar} ((y,t):ts) x \mid (x == y) = t \\
\mid \text{otherwise} = \text{substVar} ts x
\]

- extended to a Term: \( \hat{\sigma} : T(\Sigma, X) \rightarrow T(\Sigma, X) \)

\[
\text{substTerm :: Subst} \rightarrow \text{Term} \rightarrow \text{Term} \\
\text{substTerm} s (V x) = \text{substVar} s x \\
\text{substTerm} s (T f ts) = T f (\text{map} (\text{substTerm} s) ts)
\]
Basic Notions
  Signature and Terms
  Subterms and Transformed Terms

Reductions
  Identities and Reduction Relations
  Substitutions

Term Rewriting Systems
  Rewrite Rules and Term Rewriting Systems
  Properties
A rewrite rule is

- an identity \((t_1, t_2)\) where
  - \(t_1 \notin X\), i.e. \(t_1\) not a variable
  - \(\text{Var}(t_2) \subseteq \text{Var}(t_1)\), i.e. no new variables added
- denoted by \(t_1 \rightarrow t_2\)

A term rewriting system is

- a set of rewrite rules
Terminology

- $t$ reducible: $\exists s: t \rightarrow s$
- $t$ in normal form: $t$ not reducible
  - if $t$ has a uniquely determined normal form, it is denoted by $t \downarrow$
- $s$, $t$ joinable: $\exists u: s \rightarrow^* u \leftarrow^* t$
  - denoted by $s \downarrow t$
A reduction $\rightarrow$ is called

- confluent: $t_1 \leftarrow^* s \rightarrow^* t_2 \Rightarrow t_1 \downarrow t_2$
- terminating: no infinite descending chain $t_1 \rightarrow t_2 \rightarrow t_3 \rightarrow \ldots$
- normalizing: every $t$ has a normal form, i.e. at least one normal form
- convergent: confluent and terminating
- Church-Rosser: $s \leftrightarrow^* t \Leftrightarrow s \downarrow t$
Conclusions

If a reduction \( \rightarrow \) is

- confluent \( \iff \) Church-Rosser
- confluent \( \Rightarrow \) every \( t \) has at most one normal form
- terminating \( \Rightarrow \) normalizing
- terminating and confluent \( \Rightarrow \exists \) exactly one normal form

**Theorem:**
A reduction \( \rightarrow \) is terminating and confluent:
\( s \leftrightarrow^* t \Rightarrow s \downarrow= t \downarrow \)