Term Rewriting Systems

Franz Baader
Theoretical Computer Science
TU Dresden
Germany

Literature:

Term Rewriting and All That

by Franz Baader and Tobias Nipkow

Cambridge University Press

`http://www4.informatik.tu-muenchen.de/~nipkow/TRaAT/`

Dresden

Term Rewriting

What are terms?

Expressions built from variables, constant symbols, and function symbols.

E.g., Variables x, y, constant symbol 0, function symbols s (unary) and $+$ (binary, infix):

$$0, \quad x + s(0), \quad s(s(s(0)) + 0).$$

What does rewriting mean?

Rules that describe how one term can be rewritten into another one.

Examples

- Rewriting as computation mechanism: rules applied in one direction, computes normal forms
  - close relationship to functional programming
  - example: symbolic differentiation

- Rewriting as deduction mechanism: rules applied in both directions, defines equivalence classes of terms
  - equational reasoning in automated deduction
  - example: group theory

Symbolic differentiation

Arithmetic expressions that are built with the operations $+$ (binary function symbol), $*$ (binary function symbol), the indeterminates X, Y (constant symbols), and the numbers $0, 1$ (constant symbols).

Example: $((X + X) * Y) + 1$

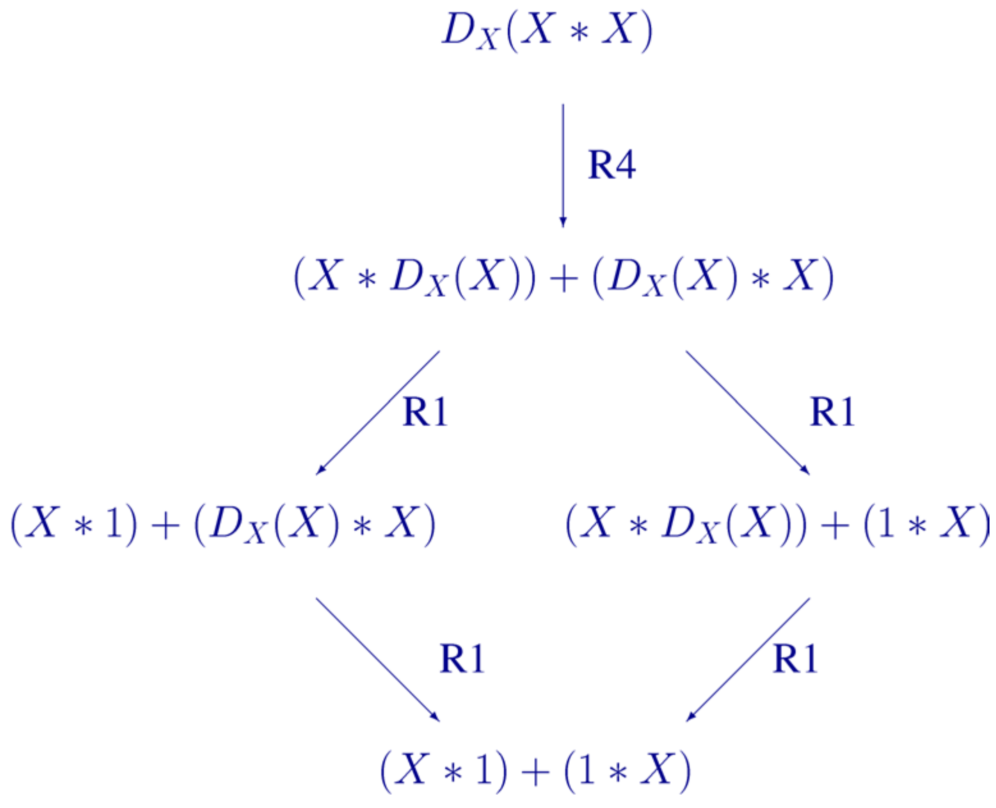
Additional (unary) function symbol D_X: partial derivative with respect to X

Rules for computing the derivative:

$$
\begin{array}{llcl}
\text{(R1)} & D_X(X) & \rightarrow & 1, \\
\text{(R2)} & D_X(Y) & \rightarrow & 0, \\
\text{(R3)} & D_X(u+v) & \rightarrow & D_X(u) + D_X(v), \\
\text{(R4)} & D_X(u*v) & \rightarrow & (u * D_X(v)) + (D_X(u) * v).
\end{array}
$$

Dresden

$$
\begin{array}{llcl}
\text{(R1)} & D_X(X) & \rightarrow & 1, \\
\text{(R2)} & D_X(Y) & \rightarrow & 0, \\
\text{(R3)} & D_X(u+v) & \rightarrow & D_X(u) + D_X(v), \\
\text{(R4)} & D_X(u * v) & \rightarrow & (u * D_X(v)) + (D_X(u) * v).
\end{array}
$$



Dresden

Important properties

of term rewriting systems

Termination:

Is it always the case that after finitely many rule applications we reach an expression to which no more rules apply (normal form)?

For the rules (R1)–(R4) this is the case.

How can we show this?

$$D_X(u * v) \quad \to \quad (u * D_X(v)) + (D_X(u) * v).$$

Non-terminating rule

$$u + v \to v + u,$$

leads to an infinite sequence of rule applications

$$(X * 1) + (1 * X) \to (1 * X) + (X * 1) \to (X * 1) + (1 * X) \to \dots$$

Dresden

Confluence:

If there are different ways of applying rules to a given term t, leading to different derived terms t_1 and t_2, can t_1 and t_2 be joined, i.e. can we always find a common term s that can be reached both from t_1 and from t_2 by rule application?

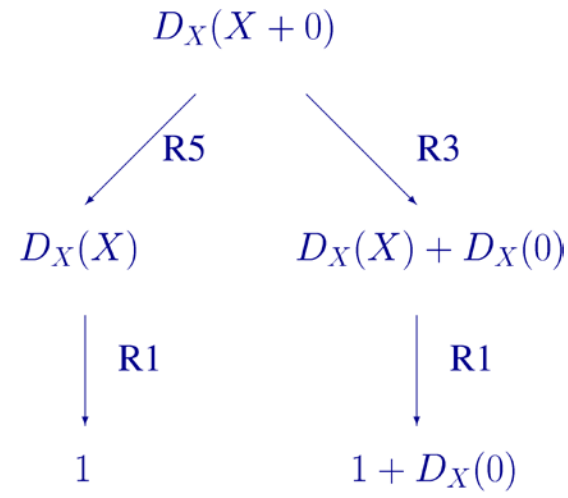
For the rules (R1)–(R4) this is the case.

How can we show this?

If we add the simplification rule

$$\text{(R5)} \quad u + 0 \rightarrow u$$

to (R1)–(R4), we lose the confluence property.



Completion: confluence can be regained by adding

$$D_X(0) \rightarrow 0$$

Group theory

Let $\circ$ be a binary function symbol, i be a unary function symbol, e be a constant symbol, and x, y, z be variable symbols.

The class of all groups is defined by the identities

(G1)	$(x \circ y) \circ z$	$\approx$	$x \circ (y \circ z)$	(associativity of $\circ$)
(G2)	$e \circ x$	$\approx$	x	(e left-unit)
(G3)	$i(x) \circ x$	$\approx$	e	(i yields left-inverse)

Identities are rewrite rules that can be applied in both direction.

Word problem

Given a set of identities E and terms s, t,
can s be rewritten into t by using the identities in E in both directions?

The identities (G1)–(G3) can be used to show that the left-inverse is also a right-inverse, i.e. e can be rewritten into $x \circ i(x)$:

$$
\begin{array}{ll}
e & \stackrel{G3}{\approx} i(x \circ i(x)) \circ (x \circ i(x)) \\
 & \stackrel{G2}{\approx} i(x \circ i(x)) \circ (x \circ (e \circ i(x))) \\
 & \stackrel{G3}{\approx} i(x \circ i(x)) \circ (x \circ ((i(x) \circ x) \circ i(x))) \\
 & \stackrel{G1}{\approx} i(x \circ i(x)) \circ ((x \circ (i(x) \circ x)) \circ i(x)) \\
 & \stackrel{G1}{\approx} i(x \circ i(x)) \circ (((x \circ i(x)) \circ x) \circ i(x)) \\
 & \stackrel{G1}{\approx} i(x \circ i(x)) \circ ((x \circ i(x)) \circ (x \circ i(x))) \\
 & \stackrel{G1}{\approx} (i(x \circ i(x)) \circ (x \circ i(x))) \circ (x \circ i(x)) \\
 & \stackrel{G3}{\approx} e \circ (x \circ i(x)) \\
 & \stackrel{G2}{\approx} x \circ i(x).
\end{array}
$$

(G1)	$(x \circ y) \circ z$	$\approx$	$x \circ (y \circ z)$
(G2)	$e \circ x$	$\approx$	x
(G3)	$i(x) \circ x$	$\approx$	e

Word problem

Given a set of identities E and terms s, t,
can s be rewritten into t by using the identities in E?

Try to solve the word problem by (uni-directional) rewriting:

$$s \xrightarrow{*} \widehat{s} = \widehat{t} \xleftarrow{*} t$$

Two problems:

- Equivalent terms can have distinct normal forms.
- Normal forms need not exist: the process of reducing a term may lead to an infinite chain of rule applications.

We will see that termination and confluence are the important properties that ensure existence and uniqueness of normal forms.