Term Rewriting Systems

Franz Baader
Theoretical Computer Science
TU Dresden
Germany

1. Motivation and basic definitions and results.
2. Equational Problems: the word problem and term rewriting
3. Termination of term rewriting systems
4. Confluence of term rewriting systems
5. Completion of term rewriting systems

Dresden

Term Rewriting Systems

Franz Baader
Theoretical Computer Science
TU Dresden
Germany

1. Motivation and basic definitions and results.
2. Equational Problems: the word problem and term rewriting
3. Termination of term rewriting systems
4. Confluence of term rewriting systems
5. Completion of term rewriting systems

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/`

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):

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

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:

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

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

$$
\begin{tikzcd}
 & D_X(X * X) \arrow[d, "\text{R4}"] & \\
 & (X * D_X(X)) + (D_X(X) * X) \arrow[dl, "\text{R1}"] \arrow[dr, "\text{R1}"] & \\
(X * 1) + (D_X(X) * X) \arrow[dr, "\text{R1}"] & & (X * D_X(X)) + (1 * X) \arrow[dl, "\text{R1}"] \\
 & (X * 1) + (1 * X) &
\end{tikzcd}
$$

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? $\quad D_X(u * v) \rightarrow (u * D_X(v)) + (D_X(u) * v).$

Non-terminating rule

$$u + v \rightarrow v + u,$$

leads to an infinite sequence of rule applications

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

Important properties

of term rewriting systems

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.

$$
\begin{tikzcd}
 & D_X(X + 0) \arrow[dl, "\text{R5}"'] \arrow[dr, "\text{R3}"] & \\
D_X(X) \arrow[d, "\text{R1}"] & & D_X(X) + D_X(0) \arrow[d, "\text{R1}"] \\
1 & & 1 + D_X(0)
\end{tikzcd}
$$

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?

Dresden

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 $x \circ i(x)$:

$$\begin{aligned}
e &\overset{G3}{\approx} i(x \circ i(x)) \circ (x \circ i(x)) \\
&\overset{G2}{\approx} i(x \circ i(x)) \circ (x \circ (e \circ i(x))) \\
&\overset{G3}{\approx} i(x \circ i(x)) \circ (x \circ ((i(x) \circ x) \circ i(x))) \\
&\overset{G1}{\approx} i(x \circ i(x)) \circ ((x \circ (i(x) \circ x)) \circ i(x)) \\
&\overset{G1}{\approx} i(x \circ i(x)) \circ (((x \circ i(x)) \circ x) \circ i(x)) \\
&\overset{G1}{\approx} i(x \circ i(x)) \circ ((x \circ i(x)) \circ (x \circ i(x))) \\
&\overset{G1}{\approx} (i(x \circ i(x)) \circ (x \circ i(x))) \circ (x \circ i(x)) \\
&\overset{G3}{\approx} e \circ (x \circ i(x)) \\
&\overset{G2}{\approx} x \circ i(x).
\end{aligned}$$

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

Dresden

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.

Dresden

Abstracts reduction systems

abstract away the internal structure of the objects that are rewritten

A pair $(A, \to)$, where

- A is an arbitrary set,
- the reduction $\to$ is a binary relation on A,

is called abstract reduction system (ARS).

$\overset{0}{\to}$	$:=$	$\{(x,x) \mid x \in A\}$	identity
$\overset{i+1}{\to}$	$:=$	$\overset{i}{\to} \circ \to$	$(i+1)$-fold composition, $i \geq 0$
$\overset{+}{\to}$	$:=$	$\bigcup_{i>0} \overset{i}{\to}$	transitive closure
$\overset{*}{\to}$	$:=$	$\overset{+}{\to} \cup \overset{0}{\to}$	reflexive transitive closure
$\overset{=}{\to}$	$:=$	$\to \cup \overset{0}{\to}$	reflexive closure
$\overset{-1}{\to}$	$:=$	$\{(y,x) \mid x \to y\}$	inverse
$\leftarrow$	$:=$	$\overset{-1}{\to}$	inverse
$\leftrightarrow$	$:=$	$\to \cup \leftarrow$	symmetric closure
$\overset{+}{\leftrightarrow}$	$:=$	$(\leftrightarrow)^+$	transitive symmetric closure
$\overset{*}{\leftrightarrow}$	$:=$	$(\leftrightarrow)^*$	reflexive transitive symmetric closure

Dresden

Let $(A, \to)$ be an ARS.

- x is reducible iff there is a y such that $x \to y$.
- x is in normal form (irreducible) iff it is not reducible.
- y is a normal form of x iff $x \xrightarrow{*} y$ and y is in normal form.
  If x has a uniquely determined normal form, the latter is denoted by $x{\downarrow}$.
- y is a direct successor of x iff $x \to y$.
- y is a successor of x iff $x \xrightarrow{+} y$
- x and y are joinable iff there is a z such that $x \xrightarrow{*} z \xleftarrow{*} y$,
  in which case we write $x \downarrow y$.

Example

Let $A := \mathbb{N} - \{0,1\}$ and

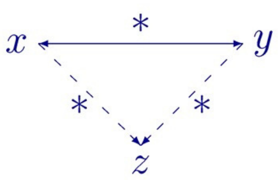
$$\to := \{(m,n) \mid m > n \text{ and } n \text{ divides } m\}.$$

1. m is in normal form iff m is prime.
2. p is a normal form of m iff p is a prime factor of m.
3. $m \downarrow n$ iff m and n are not relatively prime.
4. $\xrightarrow{+} = \to$ because $>$ and “divides” are already transitive.
5. $\xleftrightarrow{*} = A \times A$.

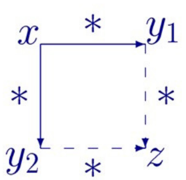
Definition

A reduction $\to$ is called

Church-Rosser	iff $x \xleftrightarrow{*} y \Rightarrow x \downarrow y$
confluent	iff $y_1 \xleftarrow{*} x \xrightarrow{*} y_2 \Rightarrow y_1 \downarrow y_2$
terminating	iff there is no infinite descending chain $a_0 \to a_1 \to \cdots$
convergent	iff it is both confluent and terminating.



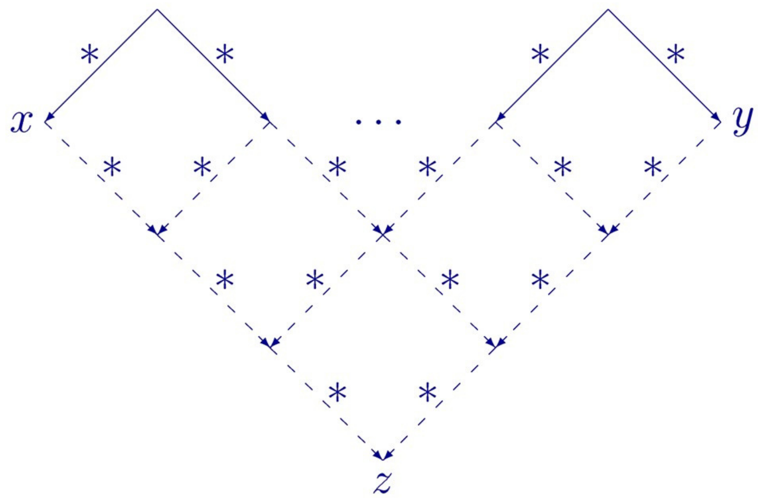
Church-Rosser



confluent

Theorem

A reduction $\to$ is confluent iff it is Church Rosser.



Theorem

If $\rightarrow$ is confluent and terminating, then

- every element x has a unique normal form $x\downarrow$.
- $x \overset{*}{\leftrightarrow} y$ iff $x\downarrow = y\downarrow$.

How can we show termination and confluence of a given ARS?

Dresden

Showing termination

by embedding into a well-founded partial order $>$

A partial order $(B, >)$ is called well-founded iff it is terminating, i.e. there is no infinite descending chain

$$b_0 > b_1 > b_2 > b_3 > \ldots$$

Theorem

Let $(A, \rightarrow)$ be an ARS. Then the following are equivalent:

- $\rightarrow$ is terminating.
- There is a well-founded partial order $(B, >)$ and a mapping $\varphi : A \rightarrow B$ such that

$$a \rightarrow a' \quad \text{implies} \quad \varphi(a) > \varphi(a').$$

Dresden

Example

Embedding into $(\mathbb{N}, >)$, which is obviously well-founded.

For strings, i.e. $A := X^*$ for some set X, the following are natural choices for mappings into $\mathbb{N}$:

1. Length: φ is defined by

   $$\varphi(w) := |w|.$$

   This mapping proves termination of all length-decreasing reductions, like

   $$uabbv \rightarrow_1 uaav,$$

   where $u, v \in A$ are arbitrary and $a, b \in X$ are fixed.

2. Letters: For each $a \in X$ define

   $$\varphi_a(w) := \text{“the number of occurrences of } a \text{ in } w\text{”}.$$

   This mapping proves termination of reductions like

   $$uav \rightarrow_2 ubv,$$

   where $u, v \in A$ are arbitrary and $a, b \in X$, $a \neq b$, are fixed.

Dresden

How about termination of $\rightarrow_1 \cup \rightarrow_2$?

Given two strict orders $(A, >_A)$ and $(B, >_B)$, the lexicographic product $>_{A\times B}$ on $A \times B$ is defined by

$$(x, y) >_{A\times B} (x', y') \;:\Leftrightarrow\; (x >_A x') \vee (x = x' \wedge y >_B y').$$

Theorem

The lexicographic product of two well-founded partial orders is again a well-founded partial order.

$$(a_0, b_0) > (a_1, b_1) > (a_2, b_2) > (a_3, b_3) > (a_4, b_4) > \cdots$$

$$a_0 \geq a_1 \geq a_2 \geq a_3 \geq a_4 \geq \cdots a_k = a_{k+1} = a_{k+2} = \cdots$$

$$b_k > b_{k+1} > b_{k+2} > \cdots$$

Dresden

Example (continued)

Embedding into $(\mathbb{N}\times\mathbb{N}, >_{\mathbb{N}\times\mathbb{N}})$, which is well-founded by the above theorem.

3. Length in first component and letters in the second:

   $\widehat{\varphi}$ is defined by

   $$\widehat{\varphi}(w) := (|w|, \varphi_a(w)).$$

   This mapping proves termination of all $\to_1 \cup \to_2$ with

   $$\begin{aligned} uabbv &\to_1 uaav, \\ uav &\to_2 ubv, \end{aligned}$$

   where $u, v \in A$ are arbitrary and $a, b \in X$, $a \neq b$, are fixed.

Well-founded induction

Let $(A, \to)$ be an ARS and P be a property of elements of A.

$$\frac{\forall x \in A.\, (\forall y \in A.\, x \xrightarrow{+} y \;\Rightarrow\; P(y)) \;\Rightarrow\; P(x)}{\forall x \in A.\, P(x)} \quad \text{(WFI)}$$

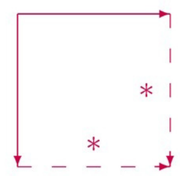
To prove $P(x)$ for all x, it suffices to prove $P(x)$ under the assumption that $P(y)$ holds for all successors y of x.

Theorem (correctness of WFI)

If $\to$ terminates then WFI holds.

Showing confluence

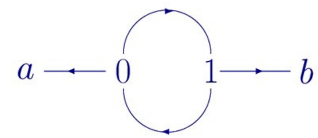
by localizing the confluence test



A reduction $\to$ is locally confluent iff

$$y_1 \leftarrow x \to y_2 \;\Rightarrow\; y_1 \downarrow y_2.$$

Confluence implies local confluence, but not vice versa.



Theorem

If $\to$ is locally confluent and terminating, then it is also confluent.

Proof by well-founded induction.

Theorem

If $\to$ is locally confluent and terminating, then it is also confluent.

We show confluence by well-founded induction using the predicate

$$P(x) \;=\; \forall y, z.\, y \xleftarrow{*} x \xrightarrow{*} z \Rightarrow y \downarrow z.$$

Obviously $\to$ is confluent if $P(x)$ holds for all x.

Show $P(x)$ under the assumption $P(t)$ for all $x \xrightarrow{+} t$:

Let $y \xleftarrow{*} x \xrightarrow{*} z$.

Case 1: $x = y$ or $x = z$ is trivial.

Case 2: $x \to y_1 \xrightarrow{*} y$ and $x \to z_1 \xrightarrow{*} z$.

$$\begin{tikzcd}
x \arrow[r] \arrow[d] \arrow[dr, phantom, "L.C."] & z_1 \arrow[r, "*"] \arrow[d, dashed, "*"] & z \arrow[dd, dashed, "*"] \\
y_1 \arrow[r, dashed, "*"] \arrow[d, "*"'] \arrow[dr, phantom, "Ind"] & u \arrow[d, dashed, "*"] \arrow[r, phantom, "Ind"] & {} \\
y \arrow[r, dashed, "*"] & v \arrow[r, dashed, "*"] & w
\end{tikzcd}$$