Rewriting theory is the study of relations and, in particular, relations on symbolic expressions (terms/trees, graphs, etc.). Relations can for instance be used to model transition systems, operational semantics, computation rules, and decide equational theories.

In this course, we will consider abstract relations or relations on untyped first-order terms, but one can also consider relations on other kinds of objects (e.g. $\lambda$-terms, graphs, etc.) or on some particular subsets of such objects (e.g. well-typed terms).

Term rewriting is based on the notion of pattern matching. Programming languages like OCaml and Haskell, or proof assistants like Agda and Coq use a restricted form of pattern matching. Programming languages like Maude \cite{13} and proof assistants like Dedukti \cite{6,1} allow the use of a more general form of pattern-matching (e.g. matching on defined symbols, matching modulo some equational theory).

Two properties will be of particular interest for us: termination (there is no infinite sequence of rewrite steps) and confluence (the order of rewrite steps does not matter).

For lecture notes in French, see for instance \cite{4}. For books in English, see for instance \cite{2,14} or \cite{17}. Finally, for some historical notes, see \cite{8}.

1 Basic definitions and results

1.1 Finite sequences or words

A word over a set $\mathcal{A}$ is a (possibly empty) finite sequence $a_1 \ldots a_n$ of elements of $\mathcal{A}$. Let $\mathcal{A}^*$ be the set of all words over $\mathcal{A}$. The empty word is denoted by $\varepsilon$ and the concatenation of two words $u$ and $v$ by their juxtaposition $uv$. We also denote by $\vec{a}$ a word $a_1 \ldots a_n$ of length $|\vec{a}| = n$. 
1.2 Terms

A signature is given by an \( \mathbb{N} \)-indexed family of (possibly empty) sets \( (F_n)_{n \in \mathbb{N}} \). An element of \( F_n \) is called a function symbol of arity \( n \). Let \( F = \bigcup_{n \in \mathbb{N}} F_n \).

**Example:** To represent arithmetical expressions, one can consider the symbols \textsf{zero} of arity 0, \textsf{suc} of arity 1 (for the successor function), \textsf{add} and \textsf{mul} of arity 2, etc.

The set \( T(F, \mathcal{V}) \) (\( T \) for short) of (untyped first-order) terms over a signature \( F \) and an infinite set \( \mathcal{V} \) of variables disjoint from \( F \) is defined inductively as follows:

- variables are terms;
- if \( f \in F_n \) and \( t_1, \ldots, t_n \) are terms, then \( f \ t_1 \ldots t_n \) is a term.

**Example:** \((\textsf{add} \ \textsf{zero} \ x)\) is a term, but neither \((x \ \textsf{zero})\) nor \textsf{add} are terms.

Let \( \mathcal{V}(t) \) (resp. \( F(t) \)) be the set of variables (resp. function symbols) occurring in \( t \). A term \( t \) is said to be ground or closed if \( \mathcal{V}(t) = \emptyset \). Let \( T(F) \) be the set of closed terms. A term is linear if no variable occurs more than once in it.

The subterms of a term \( t \) can be designated by their positions in \( t \) as follows. Given a term \( t \), the set \( \text{Pos}(t) \) of positions in \( t \) is the subset of \( \mathbb{N}^* \) (the set of words on the alphabet \( \mathbb{N} \)) inductively defined as follows:

- \( \text{Pos}(x) = \{\varepsilon\} \)
- \( \text{Pos}(f \ t_1 \ldots t_n) = \{\varepsilon\} \cup \{ip | 1 \leq i \leq n, p \in \text{Pos}(t_i)\} \)

Then, the subterm of \( t \) at position \( p \in \text{Pos}(t) \), written \( t|_p \), is defined recursively as follows:

- \( t|_\varepsilon = t \)
- \( (f \ t_1 \ldots t_n)|_ip = t_i|_p \)

Now, the term obtained by replacing in a term \( t \) its subterm at position \( p \) by a term \( u \) is written \( t[u]_p \).

**Example:** Positions and subterms in the term \((\textsf{add} \ \textsf{zero} \ (\textsf{add} \ \textsf{zero} \ x))\):

\[
\begin{array}{c}
\varepsilon: \text{add} \\
\text{1: zero} \quad \text{2: add} \\
\quad \text{1: zero} \quad \text{2: x}
\end{array}
\]
Given a signature $\mathcal{F}$, an $\mathcal{F}$-algebra $I$ is given by a non-empty set $A$ and, for each function symbol $f$ of arity $n$, a function $f_I : A^n \rightarrow A$. Given a valuation $\xi : \mathcal{V} \rightarrow A$, the interpretation of a term $t$ in $A$, written $t_\xi$, is recursively defined as follows:

- $x_\xi = \xi(x)$
- $(f t_1 \ldots t_n)_\xi = f_I(t_1\xi, \ldots, t_n\xi)$

$T$ itself is an $\mathcal{F}$-algebra by taking $f_T(t_1, \ldots, t_n) = f t_1 \ldots t_n$. A valuation $\sigma : \mathcal{V} \rightarrow T$ is called a substitution.

**Lemma 1** $(t\sigma)_\xi = t(\sigma_\xi)$ where $\sigma_\xi = \xi \circ \sigma$.

Remark: the collection of $\mathcal{F}$-algebras form a category as follows. A morphism from an $\mathcal{F}$-algebra $I = (A, (f_I)_{f \in \mathcal{F}})$ to an $\mathcal{F}$-algebra $J = (B, (f_J)_{f \in \mathcal{F}})$ is given by a function $\varphi : A \rightarrow B$ such that, for every function symbol $f$ of arity $n$, $\varphi \circ f_I = f_J \circ \varphi^n$. $T$ is the initial object in the category of $\mathcal{F}$-algebras.

### 1.3 Term rewriting

A context is a term $C$ with exactly one occurrence of a distinguished variable $\Box$. The term obtained by replacing $\Box$ by some term $u$ is written $C[u]$.

A relation on terms $R$ is:

- stable (by substitution) if, for all substitutions $\sigma$, $t\sigma Ru\sigma$ whenever $tRu$;
- stable by context or monotone if, for all contexts $C$, $C[t]RC[u]$ whenever $tRu$;
- a rewrite relation if it is both stable and monotone;
- a reduction relation if it is a terminating rewrite relation;
- a rewrite (quasi-)order if it is a rewrite relation and a (quasi-)order;
- a reduction order if it is a terminating rewrite (strict) order.

A (term) rewrite rule is a pair of terms $(l, r)$, written $l \rightarrow r$, such that $l \notin \mathcal{V}$ and $\mathcal{V}(r) \subseteq \mathcal{V}(l)$.

A term $t$ matches a term $l$ (the pattern) if there is $\sigma$ such that $t = l\sigma$. Note that a pattern may have a variable occurring more than once. Matching is decidable and of linear complexity [15].

A (term) rewrite system is a set of term rewrite rules.

**Example:** Subtraction on natural numbers can be defined by the following set of rules $\mathcal{R}_{\text{minus}}$:

```
minus zero x    →    zero
minus x zero   →    x
minus (suc x) (suc y)    →    minus x y
```
Given a rewrite system $\mathcal{R}$, let $\rightarrow_{\mathcal{R}}$ be the smallest rewrite relation containing $\mathcal{R}$: $t \rightarrow_{\mathcal{R}} u$ iff there are $p \in \text{Pos}(t)$, $l \rightarrow r \in \mathcal{R}$ and $\sigma$ such that $t|_p = l\sigma$ and $u = t[r\sigma]_p$. We say that $t$ rewrites to $u$ at position $p \in \text{Pos}(t)$, written $t \xrightarrow{p} \mathcal{R} u$, if there are $l \rightarrow r \in \mathcal{R}$ and $\sigma$ such that $t|_p = l\sigma$ and $u = t[r\sigma]_p$. Let $\geq_{\mathcal{R}} = \bigcup_{p \in \text{Pos}(t)} P_{\mathcal{R}}$. Let $\leftarrow_{\mathcal{R}}$ be the inverse of $\rightarrow_{\mathcal{R}}$, $\leftrightarrow_{\mathcal{R}}$ its symmetric closure, $\downarrow_{\mathcal{R}} = \rightarrow_{\mathcal{R}} \bowtie_{\mathcal{R}}$ the joinability relation, and $\sim_{\mathcal{R}}$ be its reflexive, symmetric and transitive closure. $\rightarrow_{\mathcal{R}}$ is (resp. locally) confluent if $\leftrightarrow_{\mathcal{R}} \subseteq \downarrow_{\mathcal{R}}$ (resp. $\leftarrow_{\mathcal{R}} \rightarrow_{\mathcal{R}} \subseteq \downarrow_{\mathcal{R}}$).

Note that the variables chosen for rules is not important: if $\rho$ is a permutation of $V$, then $\rightarrow_{\mathcal{R}} \rho = \rightarrow_{\mathcal{R}}$. By abuse of language, we often say that $\mathcal{R}$ terminates or is confluent instead of saying that $\rightarrow_{\mathcal{R}}$ terminates or is confluent.

The termination and confluence properties are undecidable for arbitrary term rewrite systems, but decidable for ground term rewrite systems [19, 18].

An immediate and complete approach to the termination of relations of the form $\rightarrow_{\mathcal{R}}$ is to define reduction orders, as suggested by the following trivial lemma:

**Lemma 2** \(\mathcal{R}\) terminates iff there is a reduction order $>$ such that $\mathcal{R} \subseteq >$.

**Proof.**

$\Leftarrow$: Since $>$ is a rewrite relation and $\mathcal{R} \subseteq >$, we have $\rightarrow_{\mathcal{R}} \subseteq >$. Thus $\rightarrow_{\mathcal{R}}$ terminates since $>$ terminates.

$\Rightarrow$: Take $> = \rightarrow_{\mathcal{R}}^+$. ■

2 **Proving termination using interpretations**

A first technique to define reduction orders, introduced by Manna & Ness in 1970 [10], is to interpret terms into a well-founded domain:

**Definition 1 (Well-founded monotone algebra)** Given a signature \(\mathcal{F}\), a relational \(\mathcal{F}\)-algebra is an algebra \(I = (A, (f_I)_{f \in \mathcal{F}})\) equipped with a relation $R$ on $A$. Such an algebra induces a relation on terms as follows: $tRfu$ iff, for all valuation $\xi : \mathcal{V} \rightarrow A$, $t\xi Ru\xi$.

We say that $(I, R)$ is well-founded if $R$ so is.

We say that $(I, R)$ is monotone if each $f_I$ is monotone wrt $R$ in every argument, that is, for all $i$,

$$f_I(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n) R f_I(x_1, \ldots, x_{i-1}, x'_i, x_{i+1}, \ldots, x_n)$$

whenever $x_i R x'_i$. 

4
Lemma 3  

- $R_I$ is stable.
- $R_I$ is monotone if $(I, R)$ is monotone.
- $R_I$ terminates if $R$ terminates.

Proof.

- Assume that $tR_Iu$. Then, $t\sigma R_I u\sigma$ since $(t\sigma)\xi = t(\sigma \xi)Ru(\sigma \xi) = (u\sigma)\xi$.
- By induction on $C$ and monotony.
- Assume that $t_0R_I t_1 R_I \ldots$. Let $a$ be any element of $A$ (algebras are non empty by assumption) and $\xi$ be the constant valuation equal to $a$. Then, $t_0\xi R t_1 \xi R \ldots$. Contradiction. ■

Theorem 1  

A rewrite system $\mathcal{R}$ terminates iff there is a well-founded monotone algebra $(I, >)$ such that $\mathcal{R} \subseteq >_I$.

Proof.

- $\Leftarrow$: Assume that $t \rightarrow_R u$. Then, there is $C$, $\sigma$ and $l \rightarrow r \in \mathcal{R}$ such that $t = C[l\sigma]$ and $u = C[r\sigma]$. Since $l >_I r$ and $>_I$ is stable by substitution, $l\sigma >_I r\sigma$. Now, since $>_I$ is stable by context, $t >_I u$. Therefore, $\rightarrow_R \subseteq >_I$ and $\rightarrow_R$ terminates.
- $\Rightarrow$: Assume that $\rightarrow_R$ terminates. Then, $(\mathcal{T}, \rightarrow_R)$ is a well-founded monotone algebra containing $\mathcal{R}$. ■

2.1 Polynomial interpretations on $\mathbb{R}$ or $\mathbb{N}$

The first-order theory of real closed fields being decidable [16, 3], one can try to interpret function symbols by polynomials over the non-negative reals but the standard ordering on non-negative reals is not well-founded. Two solutions have been proposed to get around this problem:

- Take the standard ordering on reals but consider strictly extensive polynomials only, that is, such that $t_{\mathbb{R}}(\bar{x}) > x_i$ for every $i$ [7].
- Consider, for some fixed $\delta > 0$, the ordering $x >_\delta y$ iff $x \geq y + \delta$ [9].

Now, assume given a formal polynomial $P_t \in \mathbb{R}[X_1, \ldots, X_n]$ for each $f \in \mathcal{F}_n$. Then, every term $t$ with $\mathcal{V}(t) \subseteq \{X_1, \ldots, X_n\}$ can be interpreted by a polynomial $P_t(X_1, \ldots, X_n) \in \mathbb{R}[X_1, \ldots, X_n]$ as follows:

- $P_{X_i}(X_1, \ldots, X_n) = X_i$,
- $P_t(P_{t_1}(X_1, \ldots, X_n), \ldots, P_{t_n}(X_1, \ldots, X_n))$, that is, $P_t$ with $X_i$ replaced by $P_{t_i}$.
Then, the condition $l > r$ reduces to $P_l - P_r > 0$.

Now, even though the resolution of diophantine (in)equations is undecidable in general \[11, 12\], trying to interpret function symbols by polynomials over $\mathbb{N}$ with small degree and coefficients gives useful results in practice.

In this case, the monotony condition can be reduced to a positivity check too since, in $\mathbb{N}$, $x > y$ is equivalent to $x - y - 1 \geq 0$. Moreover, it suffices to check that $P_l(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) - P_l(x_1, \ldots, x_n) - 1 \geq 0$.

In both cases, a simple decidable condition for checking the non-negativity of a polynomial is to check that every coefficient is non-negative.

**Example:** The rewrite system of Example 1.3 can be proved terminating by using the following polynomial interpretation on $\mathbb{N}$: $\text{zero} = 0$, $\text{suc}(x) = x + 1$, $\text{minus}(x, y) = x + y + 1$.

### 2.2 Affine interpretations on $\mathbb{N}^d$

Another, more powerful technique, is to interpret terms as vectors ($A = \mathbb{N}^d$ for some $d \in \mathbb{N}^*$) and function symbols as affine transformations ($f_I(x_1, \ldots, x_n) = \sum_{i=1}^{n} M_i x_i + v$ where $M_i \in \mathbb{N}^{d \times d}$ and $v \in \mathbb{N}^d$). Several orderings can be used on vectors. The mostly used is $x > y$ if $x_1 > y_1$ and, for all $i > 1$, $x_i \geq y_i$.

**References**


