A relation $\to$ is called

terminating, if there is no infinite descending chain $b_0 \to b_1 \to b_2 \to \ldots$.

normalizing, if every $b \in A$ has a normal form.

Lemma 1.10 *If $\to$ is terminating, then it is normalizing.*

Note: The reverse implication does not hold.

Confluence

Let $(A, \to)$ be a rewrite system.

b and $c \in A$ are *joinable*, if there is an a such that $b \to^* a \; {}^*\!\leftarrow c$.
Notation: $b \downarrow c$.

The relation $\to$ is called

Church-Rosser, if $b \leftrightarrow^* c$ implies $b \downarrow c$.

confluent, if $b \; {}^*\!\leftarrow a \to^* c$ implies $b \downarrow c$.

locally confluent, if $b \leftarrow a \to c$ implies $b \downarrow c$.

convergent, if it is confluent and terminating.

For a rewrite system $(M, \to)$ consider a sequence of elements a_i that are pairwise connected by the symmetric closure, i.e., $a_1 \leftrightarrow a_2 \leftrightarrow a_3 \ldots \leftrightarrow a_n$. We say that a_i is a *peak* in such a sequence, if actually $a_{i-1} \leftarrow a_i \to a_{i+1}$.

Theorem 1.11 *The following properties are equivalent:*

(i) $\to$ *has the Church-Rosser property.*

(ii) $\to$ *is confluent.*

Proof. (i)$\Rightarrow$(ii): trivial.

(ii)$\Rightarrow$(i): by induction on the number of peaks in
the derivation $b \leftrightarrow^* c$. $\square$

Lemma 1.12 *If $\to$ is confluent, then every element has at most one normal form.*

Proof. Suppose that some element $a \in A$ has normal forms b and c, then $b \;{}^*\!\!\leftarrow a \rightarrow^* c$. If $\rightarrow$ is confluent, then $b \rightarrow^* d \;{}^*\!\!\leftarrow c$ for some $d \in A$. Since b and c are normal forms, both derivations must be empty, hence $b \rightarrow^0 d \;{}^0\!\!\leftarrow c$, so b, c, and d must be identical. $\square$

Corollary 1.13 *If $\rightarrow$ is normalizing and confluent, then every element b has a unique normal form.*

Proposition 1.14 *If $\rightarrow$ is normalizing and confluent, then $b \leftrightarrow^* c$ if and only if $b\!\downarrow = c\!\downarrow$.*

Proof. Either using Thm. 1.11 or directly by induction on the length of the derivation of $b \leftrightarrow^* c$. $\square$

Confluence and Local Confluence

Theorem 1.15 ("Newman's Lemma") *If a terminating relation $\rightarrow$ is locally confluent, then it is confluent.*

Proof. Let $\rightarrow$ be a terminating and locally confluent relation. Then $\rightarrow^+$ is a well-founded ordering. Define $Q(a) \;\Leftrightarrow\; \big(\forall b, c : b \;{}^*\!\!\leftarrow a \rightarrow^* c \Rightarrow b \downarrow c\big)$.

We prove $Q(a)$ for all $a \in A$ by well-founded induction over $\rightarrow^+$:

Case 1: $b \;{}^0\!\!\leftarrow a \rightarrow^* c$: trivial.

Case 2: $b \;{}^*\!\!\leftarrow a \rightarrow^0 c$: trivial.

Case 3: $b \;{}^*\!\!\leftarrow b' \leftarrow a \rightarrow c' \rightarrow^* c$: use local confluence, then use the induction hypothesis. $\square$

2 Propositional Logic

Propositional logic

- logic of truth values
- decidable (but **NP**-complete)
- can be used to describe functions over a finite domain
- industry standard for many analysis/verification tasks
- growing importance for discrete optimization problems (Automated Reasoning II)

2.1 Syntax

- propositional variables
- logical connectives
  $\Rightarrow$ Boolean connectives and constants

Propositional Variables

Let Σ be a set of *propositional variables* also called the *signature* of the (propositional) logic.

We use letters P, Q, R, S, to denote propositional variables.

Propositional Formulas

$\mathrm{PROP}(\Sigma)$ is the set of propositional formulas over Σ inductively defined as follows:

$$
\begin{array}{rcll}
\phi, \psi & ::= & \bot & \text{(falsum)} \\
& \mid & \top & \text{(verum)} \\
& \mid & P, \quad P \in \Sigma & \text{(atomic formula)} \\
& \mid & \neg\phi & \text{(negation)} \\
& \mid & (\phi \wedge \psi) & \text{(conjunction)} \\
& \mid & (\phi \vee \psi) & \text{(disjunction)} \\
& \mid & (\phi \rightarrow \psi) & \text{(implication)} \\
& \mid & (\phi \leftrightarrow \psi) & \text{(equivalence)}
\end{array}
$$

Notational Conventions

As a notational convention we assume that $\neg$ binds strongest, so $\neg P \vee Q$ is actually a shorthand for $(\neg P) \vee Q$. For all other logical connectives we will explicitly put parenthesis when needed. From the semantics we will see that $\wedge$ and $\vee$ are associative and commutative. Therefore instead of $((P \wedge Q) \wedge R)$ we simply write $P \wedge Q \wedge R$.

Automated reasoning is very much formula manipulation. In order to precisely represent the manipulation of a formula, we introduce positions.

Formula Manipulation

A *position* is a word over $\mathbb{N}$. The set of positions of a formula ϕ is inductively defined by

$$\begin{array}{rcl} \mathrm{pos}(\phi) & := & \{\epsilon\} \text{ if } \phi \in \{\top, \bot\} \text{ or } \phi \in \Sigma \\ \mathrm{pos}(\neg\phi) & := & \{\epsilon\} \cup \{1p \mid p \in \mathrm{pos}(\phi)\} \\ \mathrm{pos}(\phi \circ \psi) & := & \{\epsilon\} \cup \{1p \mid p \in pos(\phi)\} \cup \{2p \mid p \in \mathrm{pos}(\psi)\} \end{array}$$

where $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

The prefix order $\leq$ on positions is defined by $p \leq q$ if there is some p' such that $pp' = q$.

Note that the prefix order is partial, e.g., the positions 12 and 21 are not comparable, they are “parallel”, see below.

By $<$ we denote the strict part of $\leq$, i.e., $p < q$ if $p \leq q$ but not $q \leq p$. By $\parallel$ we denote incomparable positions, i.e., $p \parallel q$ if neither $p \leq q$, nor $q \leq p$. Then we say that p is *above* q if $p \leq q$, p is *strictly above* q if $p < q$, and p and q are *parallel* if $p \parallel q$.

The *size* of a formula ϕ is given by the cardinality of $\mathrm{pos}(\phi)$: $|\phi| := |\mathrm{pos}(\phi)|$.

The *subformula* of ϕ at position $p \in \mathrm{pos}(\phi)$ is recursively defined by $\phi|_\epsilon := \phi$ and $(\phi_1 \circ \phi_2)|_{ip} := \phi_i|_p$ where $i \in \{1, 2\}$, $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

Finally, the *replacement* of a subformula at position $p \in \mathrm{pos}(\phi)$ by a formula ψ is recursively defined by

$$\begin{array}{rcl} \phi[\psi]_\epsilon & := & \psi \\ (\neg\phi)[\psi]_{1p} & := & \neg(\phi[\psi]_p) \\ (\phi_1 \circ \phi_2)[\psi]_{1p} & := & (\phi_1[\psi]_p \circ \phi_2) \\ (\phi_1 \circ \phi_2)[\psi]_{2p} & := & (\phi_1 \circ \phi_2[\psi]_p) \end{array}$$

where $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

Example 2.1 *The set of positions for the formula* $\phi = (A \wedge B) \rightarrow (A \vee B)$ *is* $\operatorname{pos}(\phi) = \{\epsilon, 1, 11, 12, 2, 21, 22\}$. *The subformula at position* 22 *is* B, $\phi|_{22} = B$ *and replacing this formula by* $A \leftrightarrow B$ *results in* $\phi[A \leftrightarrow B]_{22} = (A \wedge B) \rightarrow (A \vee (A \leftrightarrow B))$.

A further prerequisite for efficient formula manipulation is the polarity of a subformula ψ of ϕ. The polarity determines the number of "negations" starting from ϕ down to ψ. It is 1 for an even number along the path, -1 for an odd number and 0 if there is at least one equivalence connective along the path.

The *polarity* of a subformula ψ of ϕ at position p, $i \in \{1, 2\}$ is recursively defined by

$$
\begin{array}{rcl}
\operatorname{pol}(\phi, \epsilon) & := & 1 \\
\operatorname{pol}(\neg\phi, 1p) & := & -\operatorname{pol}(\phi, p) \\
\operatorname{pol}(\phi_1 \circ \phi_2, ip) & := & \operatorname{pol}(\phi_i, p) \text{ if } \circ \in \{\wedge, \vee\} \\
\operatorname{pol}(\phi_1 \rightarrow \phi_2, 1p) & := & -\operatorname{pol}(\phi_2, p) \\
\operatorname{pol}(\phi_1 \rightarrow \phi_2, 2p) & := & \operatorname{pol}(\phi_2, p) \\
\operatorname{pol}(\phi_1 \leftrightarrow \phi_2, ip) & := & 0
\end{array}
$$

Example 2.2 *We reuse the formula* $\phi = (A \wedge B) \rightarrow (A \vee B)$ *Then* $\operatorname{pol}(\phi, 1) = \operatorname{pol}(\phi, 11) = -1$ *and* $\operatorname{pol}(\phi, 2) = \operatorname{pol}(\phi, 22) = 1$. *For the formula* $\phi' = (A \wedge B) \leftrightarrow (A \vee B)$ *we get* $\operatorname{pol}(\phi', \epsilon) = 1$ *and* $\operatorname{pol}(\phi', p) = 0$ *for all other* $p \in \operatorname{pos}(\phi')$, $p \neq \epsilon$.