1.5 Complexity Theory Prerequisites

A *decision problem* is a subset $L \subseteq \Sigma^*$ for some fixed finite alphabet Σ.

The function $\mathrm{chr}(L, x)$ denotes the *characteristic function* for some decision problem L and is defined by $\mathrm{chr}(L, u) = 1$ if $u \in L$ and $\mathrm{chr}(L, u) = 0$ otherwise.

P and NP

A decision problem is called *solvable in polynomial-time* if its characteristic function can be computed in polynomial-time. The class P denotes all polynomial-time decision problems.

We say that a decision problem L is in *NP* if there is a predicate $Q(x, y)$ and a polynomial $p(n)$ such that for all $u \in \Sigma^*$ we have

(i) $u \in L$ if and only if there is a $v \in \Sigma^*$ with $|v| \leq p(|u|)$ and $Q(u, v)$ holds, and

(ii) the predicate Q is in P.

Reducibility, NP-Hardness, NP-Completeness

A decision problem L is *polynomial-time reducible* to a decision problem L' if there is a function $g \in \mathrm{P}$ such that for all $u \in \Sigma^*$ we have $u \in L$ iff $g(u) \in L'$.

For example, if L is reducible to L' and $L' \in \mathrm{P}$ then $L \in \mathrm{P}$.

A decision problem is *NP-hard* if every problem in NP is polynomial-time reducible to it.

A decision problem is *NP-complete* if it is NP-hard and in NP.

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 (e. g., model checking),
- growing importance for discrete optimization problems

2.1 Syntax

- propositional variables
- logical connectives
  $\Rightarrow$ Boolean combinations

Propositional Variables

Let Π be a set of *propositional variables*.

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

Propositional Formulas

F_Π is the set of propositional formulas over Π defined inductively as follows:

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

Notational Conventions

As a notational convention we assume that $\neg$ binds strongest, and we remove outermost parentheses, so $\neg P \vee Q$ is actually a shorthand for $((\neg P) \vee Q)$. For all other logical connectives we will use parentheses 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 $P \wedge Q \wedge R$.

Formula Manipulation

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

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

$$\begin{aligned} \mathrm{pos}(F) &:= \{\varepsilon\} \text{ if } F \in \{\top, \bot\} \text{ or } F \in \Pi \\ \mathrm{pos}(\neg F) &:= \{\varepsilon\} \cup \{\, 1p \mid p \in \mathrm{pos}(F) \,\} \\ \mathrm{pos}(F \circ G) &:= \{\varepsilon\} \cup \{\, 1p \mid p \in \mathrm{pos}(F) \,\} \cup \{\, 2p \mid p \in \mathrm{pos}(G) \,\} \\ &\quad \text{where } \circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}. \end{aligned}$$

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 $\leq$, that is, $p < q$ if $p \leq q$ but not $q \leq p$.

By $\parallel$ we denote incomparable positions, that is, $p \parallel q$ if neither $p \leq q$ nor $q \leq p$.

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 F is given by the cardinality of $\mathrm{pos}(F)$: $|F| := |\mathrm{pos}(F)|$.

The *subformula* of F at position $p \in \mathrm{pos}(F)$ is recursively defined by

$$\begin{aligned} F|_\varepsilon &:= F \\ (\neg F)|_{1p} &:= F|_p \\ (F_1 \circ F_2)|_{ip} &:= F_i|_p \quad \text{where } i \in \{1, 2\} \\ &\qquad\qquad \text{and } \circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}. \end{aligned}$$

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

$$\begin{aligned} F[G]_\varepsilon &:= G \\ (\neg F)[G]_{1p} &:= \neg(F[G]_p) \\ (F_1 \circ F_2)[G]_{1p} &:= (F_1[G]_p \circ F_2) \\ (F_1 \circ F_2)[G]_{2p} &:= (F_1 \circ F_2[G]_p) \\ &\qquad \text{where } \circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}. \end{aligned}$$

Example 2.1 *The set of positions for the formula* $F = (A \wedge B) \rightarrow (A \vee B)$ *is* $\mathrm{pos}(F) = \{\varepsilon, 1, 11, 12, 2, 21, 22\}$.

The subformula at position 22 *is* $F|_{22} = B$ *and replacing this formula by* $A \leftrightarrow B$ *results in* $F[A \leftrightarrow B]_{22} = (A \wedge B) \rightarrow (A \vee (A \leftrightarrow B))$.

Polarities

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

The *polarity* of a subformula $G = F|_p$ at position p is $\mathrm{pol}(F, p)$, where pol is recursively defined by

$$\begin{aligned} \mathrm{pol}(F, \varepsilon) &:= 1 \\ \mathrm{pol}(\neg F, 1p) &:= -\mathrm{pol}(F, p) \\ \mathrm{pol}(F_1 \circ F_2, ip) &:= \mathrm{pol}(F_i, p) \text{ if } \circ \in \{\wedge, \vee\} \\ \mathrm{pol}(F_1 \rightarrow F_2, 1p) &:= -\mathrm{pol}(F_1, p) \\ \mathrm{pol}(F_1 \rightarrow F_2, 2p) &:= \mathrm{pol}(F_2, p) \\ \mathrm{pol}(F_1 \leftrightarrow F_2, ip) &:= 0 \end{aligned}$$

Example 2.2 *Let* $F = (A \rightarrow B) \rightarrow (A \vee \neg B)$. *Then* $\mathrm{pol}(F, 1) = \mathrm{pol}(F, 12) = \mathrm{pol}(F, 21) = -1$ *and* $\mathrm{pol}(F, \varepsilon) = \mathrm{pol}(F, 2) = \mathrm{pol}(F, 22) = \mathrm{pol}(F, 11) = 1$.

For the formula $F' = (A \wedge B) \leftrightarrow (A \vee B)$ *we get* $\mathrm{pol}(F', \varepsilon) = 1$ *and* $\mathrm{pol}(F', p) = 0$ *for all* $p \in \mathrm{pos}(F')$ *different from* ε.