3.13.6 Lemma (Lifting)

Let $D \lor L$ and $C \lor L'$ be variable-disjoint clauses and $\sigma$ a grounding substitution for $C \lor L$ and $D \lor L'$. If there is a superposition left inference
\[
(N \cup \{(D \lor L)\sigma, (C \lor L')\sigma\}) \Rightarrow_{\text{SUP}} (N \cup \{(D \lor L)\sigma, (C \lor L')\sigma\} \cup \{D\sigma \lor C\sigma\})
\]
and if $\text{sel}((D \lor L)\sigma) = \text{sel}((D \lor L))\sigma$, $\text{sel}((C \lor L')\sigma) = \text{sel}((C \lor L'))\sigma$, then there exists a mgu $\tau$ such that
\[
(N \cup \{D \lor L, C \lor L'\}) \Rightarrow_{\text{SUP}} (N \cup \{D \lor L, C \lor L'\} \cup \{(D \lor C)\tau\}).
\]

Let $C \lor L \lor L'$ be a clause and $\sigma$ a grounding substitution for $C \lor L \lor L'$. If there is a factoring inference
\[
(N \cup \{(C \lor L \lor L')\sigma\}) \Rightarrow_{\text{SUP}} (N \cup \{(C \lor L \lor L')\sigma\} \cup \{(C \lor L)\sigma\})
\]
and if $\text{sel}((C \lor L \lor L')\sigma) = \text{sel}((C \lor L \lor L'))\sigma$, then there exists a mgu $\tau$ such that
\[
(N \cup \{C \lor L \lor L'\}) \Rightarrow_{\text{SUP}} (N \cup \{C \lor L \lor L'\} \cup \{(C \lor L)\tau\}).
\]
3.13.7 Example (First-Order Reductions are not Liftable)

Consider the two clauses $P(x) \lor Q(x)$, $P(g(y))$ and grounding substitution \{ $x \mapsto g(a), y \mapsto a$ \}. Then $P(g(y))_\sigma$ subsumes $(P(x) \lor Q(x))_\sigma$ but $P(g(y))$ does not subsume $P(x) \lor Q(x)$. For all other reduction rules similar examples can be constructed.
3.13.8 Lemma (Soundness and Completeness)
First-Order Superposition is sound and complete.

3.13.9 Lemma (Redundant Clauses are Obsolete)
If a clause set \( N \) is unsatisfiable, then there is a derivation \( N \Rightarrow^{\ast}_{\text{SUP}} N' \) such that \( \bot \in N' \) and no clause in the derivation of \( \bot \) is redundant.

3.13.10 Lemma (Model Property)
If \( N \) is a saturated clause set and \( \bot \not\in N \) then \( \text{ground}(\Sigma, N)_{\mathcal{I}} \models N \).
Equational Logic

From now on First-order Logic is considered with equality. In this chapter, I investigate properties of a set of unit equations. For a set of unit equations I write $E$.

Full first-order clauses with equality are studied in the chapter on first-order superposition with equality. I recall certain definitions from Section 1.6 and Chapter 3.
The main reasoning problem considered in this chapter is given a set of unit equations $E$ and an additional equation $s \approx t$, does $E \models s \approx t$ hold?

As usual, all variables are implicitly universally quantified. The idea is to turn the equations $E$ into a convergent term rewrite system (TRS) $R$ such that the above problem can be solved by checking identity of the respective normal forms: $s \downarrow_R = t \downarrow_R$.

Showing $E \models s \approx t$ is as difficult as proving validity of any first-order formula, see the section on complexity.
An *equivalence* relation $\sim$ on a term set $T(\Sigma, \mathcal{X})$ is a reflexive, transitive, symmetric binary relation on $T(\Sigma, \mathcal{X})$ such that if $s \sim t$ then $\text{sort}(s) = \text{sort}(t)$.

Two terms $s$ and $t$ are called *equivalent*, if $s \sim t$.

An equivalence $\sim$ is called a *congruence* if $s \sim t$ implies $u[s] \sim u[t]$, for all terms $s, t, u \in T(\Sigma, \mathcal{X})$. Given a term $t \in T(\Sigma, \mathcal{X})$, the set of all terms equivalent to $t$ is called the *equivalence class of $t$ by $\sim$*, denoted by $[t]_{\sim} := \{ t' \in T(\Sigma, \mathcal{X}) \mid t' \sim t \}$.
If the matter of discussion does not depend on a particular equivalence relation or it is unambiguously known from the context, \([t]\) is used instead of \([t]_\sim\). The above definition is equivalent to Definition 3.2.3.

The set of all equivalence classes in \(T(\Sigma, \mathcal{X})\) defined by the equivalence relation is called a *quotient by* \(\sim\), denoted by \(T(\Sigma, \mathcal{X})|_\sim := \{[t] \mid t \in T(\Sigma, \mathcal{X})\}\). Let \(E\) be a set of equations then \(\sim_E\) denotes the smallest congruence relation “containing” \(E\), that is, \((l \approx r) \in E\) implies \(l \sim_E r\). The equivalence class \([t]_{\sim_E}\) of a term \(t\) by the equivalence (congruence) \(\sim_E\) is usually denoted, for short, by \([t]_E\). Likewise, \(T(\Sigma, \mathcal{X})|_E\) is used for the quotient \(T(\Sigma, \mathcal{X})|_{\sim_E}\) of \(T(\Sigma, \mathcal{X})\) by the equivalence (congruence) \(\sim_E\).
### 4.1.1 Definition (Rewrite Rule, Term Rewrite System)

A *rewrite rule* is an equation $l \approx r$ between two terms $l$ and $r$ so that $l$ is not a variable and $\text{vars}(l) \supseteq \text{vars}(r)$. A *term rewrite system* $R$, or a TRS for short, is a set of rewrite rules.

### 4.1.2 Definition (Rewrite Relation)

Let $E$ be a set of (implicitly universally quantified) equations, i.e., unit clauses containing exactly one positive equation. The *rewrite relation* $\rightarrow_E \subseteq T(\Sigma, \mathcal{X}) \times T(\Sigma, \mathcal{X})$ is defined by

$$s \rightarrow_E t \quad \text{iff} \quad \text{there exist } (l \approx r) \in E, p \in \text{pos}(s), \text{ and matcher } \sigma, \text{ so that } s|_p = l\sigma \text{ and } t = s[r\sigma]_p.$$
Note that in particular for any equation \( l \approx r \in E \) it holds \( l \rightarrow_E r \), so the equation can also be written \( l \rightarrow r \in E \).

Often \( s = t \downarrow_R \) is written to denote that \( s \) is a normal form of \( t \) with respect to the rewrite relation \( \rightarrow_R \). Notions \( \rightarrow^0_R, \rightarrow^+_R, \rightarrow^*_R, \leftrightarrow^*_R \), etc. are defined accordingly, see Section 1.6.
An instance of the left-hand side of an equation is called a redex (reducible expression). Contracting a redex means replacing it with the corresponding instance of the right-hand side of the rule.

A term rewrite system $R$ is called convergent if the rewrite relation $\rightarrow_R$ is confluent and terminating. A set of equations $E$ or a TRS $R$ is terminating if the rewrite relation $\rightarrow_E$ or $\rightarrow_R$ has this property. Furthermore, if $E$ is terminating then it is a TRS.

A rewrite system is called right-reduced if for all rewrite rules $l \rightarrow r$ in $R$, the term $r$ is irreducible by $R$. A rewrite system $R$ is called left-reduced if for all rewrite rules $l \rightarrow r$ in $R$, the term $l$ is irreducible by $R \setminus \{l \rightarrow r\}$. A rewrite system is called reduced if it is left- and right-reduced.
4.1.3 Lemma (Left-Reduced TRS)
Left-reduced terminating rewrite systems are convergent. Convergent rewrite systems define unique normal forms.

4.1.4 Lemma (TRS Termination)
A rewrite system $R$ terminates iff there exists a reduction ordering $\succ$ so that $l \succ r$, for each rule $l \rightarrow r$ in $R$. 