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 Chapter 5. 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 Section 3.15.

For example consider the equational ground clauses $E = \{g(a) \approx b, a \approx b\}$ over a signature consisting of the constants $a$, $b$ and unary function $g$, all defined over some unique sort. Then for all algebras $A$ satisfying $E$, all ground terms over $a$, $b$, and $g$, are mapped to the same domain element. In particular, it holds $E \models g(b) \approx b$. Now the idea is to turn $E$ into a convergent term rewrite system $R$ such that $g(b) \downarrow_R = b \downarrow_R$. To this end, the equations in $E$ are oriented, e.g., a first guess might be the TRS $R_0 = \{g(a) \rightarrow b, a \rightarrow b\}$. For $R_0$ we get $g(b) \downarrow_{R_0} = g(b), b \downarrow_{R_0} = b$, so not the desired result. The TRS $R_0$ is not confluent an all ground terms, because $g(a) \rightarrow_{R_0} b$ and $g(a) \rightarrow_{R_0} g(b)$, but $b$ and $g(b)$ are $R_0$ normal forms. This problem can be repaired by adding the extra rule $g(b) \rightarrow b$ and this process is called completion and is studied in this chapter. Now the extended rewrite system $R_1 = \{g(a) \rightarrow b, a \rightarrow b, g(b) \rightarrow b\}$ is convergent and $g(b) \downarrow_{R_1} = b \downarrow_{R_1} = b$. Termination can be shown by using a KBO (or LPO) with precedence $g \succ a \succ b$. Then the left hand sides of the rules are strictly larger than the right hand sides. Actually, $R_1$ contains some redundancy, even removing the first rewrite rule $g(a) \rightarrow b$ from $R_1$ does not violate confluence. Detecting redundant rules is also discussed in this chapter.

**Definition 4.0.1** (Equivalence Relation, Congruence Relation). An equivalence relation $\sim$ on a term set $T(\Sigma, X)$ 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 \( T(\Sigma, \mathcal{X})|_E \) of \( T(\Sigma, \mathcal{X}) \) by the equivalence (congruence) \( \sim_E \).

### 4.1 Term Rewrite System

I instantiate the abstract rewrite systems of Section 1.6 with first-order terms. The main difference is that rewriting takes not only place at the top position of a term, but also at inner positions.

**Definition 4.1.1** (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.

**Definition 4.1.2** (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

\[
 s \rightarrow_E t \quad \text{iff} \quad \text{there exist} \ (l \approx r) \in E, \ p \in \text{pos}(s), \\
 \text{and matcher} \, \sigma, \ \text{so that} \, \sigma_p = l \sigma \ \text{and} \, t = s[r\sigma]_p.
\]

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 \( \rightarrow_R^0, \rightarrow_R^+, \rightarrow_R^\ast, \leftrightarrow_R, \) 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.

**Lemma 4.1.3** (Left-Reduced TRS). Left-reduced terminating rewrite systems are convergent. Convergent rewrite systems define unique normal forms.

**Lemma 4.1.4** (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 $R$.

### 4.1.1 E-Algebras

Let $E$ be a set of universally quantified equations. A model $A$ of $E$ is also called an E-algebra. If $E \models \forall \bar{x}(s \approx t)$, i.e., $\forall \bar{x}(s \approx t)$ is valid in all $E$-algebras, this is also denoted with $s \approx^E t$. The goal is to use the rewrite relation $\rightarrow^E$ to express the semantic consequence relation syntactically: $s \approx^E t$ if and only if $s \leftrightarrow^* E t$.

Let $E$ be a set of (well-sorted) equations over $T(\Sigma, \mathcal{X})$ where all variables are implicitly universally quantified. The following inference system allows to derive consequences of $E$:

- **Reflexivity** $E \Rightarrow_E E \cup \{t \approx t\}$
- **Symmetry** $E \cup \{t \approx t'\} \Rightarrow_E E \cup \{t \approx t'\} \cup \{t' \approx t\}$
- **Transitivity** $E \cup \{t \approx t', t' \approx t''\} \Rightarrow_E E \cup \{t \approx t', t' \approx t''\} \cup \{t \approx t''\}$
- **Congruence** $E \cup \{t_1 \approx t_1', \ldots, t_n \approx t_n'\} \Rightarrow_E E \cup \{t_1 \approx t_1', \ldots, t_n \approx t_n'\} \cup \{f(t_1, \ldots, t_n) \approx f(t_1', \ldots, t_n')\}$ for any function $f : \text{sort}(t_1) \times \ldots \times \text{sort}(t_n) \rightarrow S$ for some $S$
- **Instance** $E \cup \{t \approx t'\} \Rightarrow_E E \cup \{t \approx t'\} \cup \{t \sigma \approx t' \sigma\}$ for any well-sorted substitution $\sigma$

**Lemma 4.1.5** (Equivalence of $\leftrightarrow^* E$ and $\Rightarrow^*_E$). The following properties are equivalent:

1. $s \leftrightarrow^*_E t$
2. $E \Rightarrow^*_E s \approx t$ is derivable.

where $E \Rightarrow^*_E s \approx t$ is an abbreviation for $E \Rightarrow^*_E E'$ and $s \approx t \in E'$.

**Proof.** (i)$\Rightarrow$(ii): $s \leftrightarrow^*_E t$ implies $E \Rightarrow^*_E s \approx t$ by induction on the depth of the position where the rewrite rule is applied; then $s \leftrightarrow^*_E t$ implies $E \Rightarrow^*_E s \approx t$ by induction on the number of rewrite steps in $s \leftrightarrow^*_E t$.

(ii)$\Rightarrow$(i): By induction on the size (number of symbols) of the derivation for $E \Rightarrow^*_E s \approx t$. \qed
Corollary 4.1.6 (Convergence of $E$). If a set of equations $E$ is convergent then $s \approx_E t$ if and only if $s \leftrightarrow^* t$ if and only if $s \downarrow_E = t \downarrow_E$.

Corollary 4.1.7 (Decidability of $\approx_E$). If a set of equations $E$ is finite and convergent then $\approx_E$ is decidable.

The above Lemma 4.1.5 shows equivalence of the syntactically defined relations $\leftrightarrow_E^*$ and $\Rightarrow_E^*$. What is missing, in analogy to Herbrand's theorem for first-order logic without equality Theorem 3.5.5, is a semantic characterization of the relations by a particular algebra.

Definition 4.1.8 (Quotient Algebra). For sets of unit equations this is a quotient algebra: Let $X$ be a set of variables. For $t \in T(\Sigma, \mathcal{X})$ let $[t] = \{ t' \in T(\Sigma, \mathcal{X}) \mid E \Rightarrow_E^* t \approx t' \}$ be the congruence class of $t$. Define a $\Sigma$-algebra $I_E$, called the quotient algebra, technically $T(\Sigma, \mathcal{X})/E$, as follows: $S^{I_E} = \{ [t] \mid t \in T_S(\Sigma, \mathcal{X}) \}$ for all sorts $S$ and $f^{I_E}([t_1], \ldots, [t_n]) = [f(t_1, \ldots, t_n)]$ for $f : \text{sort}(t_1) \times \cdots \times \text{sort}(t_n) \to T \in \Omega$ for some sort $T$.

Lemma 4.1.9 ($I_E$ is an $E$-algebra). $I_E = T(\Sigma, \mathcal{X})/E$ is an $E$-algebra.

Proof. Firstly, all functions $f^{I_E}$ are well-defined: if $[t_i] = [t_i']$, then $[f(t_1, \ldots, t_n)] = [f(t_1', \ldots, t_n')]$. This follows directly from the Congruence rule for $\Rightarrow^*$.

Secondly, let $\forall x_1 \ldots x_n (s \approx t)$ be an equation in $E$. Let $\beta$ be an arbitrary assignment. It has to be shown that $I_E(\beta)(\forall \bar{x}(s \approx t)) = 1$, or equivalently, that $I_E(\gamma)(s) = I_E(\gamma)(t)$ for all $\gamma = \beta[x_i \mapsto [t_i] \mid 1 \leq i \leq n]$ with $[t_i] \in \text{sort}(x_i)^{I_E}$. Let $\sigma = \{ x_1 \mapsto t_1, \ldots, x_n \mapsto t_n \}$, with $t_i \in T_{\text{sort}(x_i)}(\Sigma, \mathcal{X})$, then $s \sigma = t \sigma \in I_E(\gamma)(s)$ and $t \sigma \in I_E(\gamma)(t)$. By the Instance rule, $E \Rightarrow^* s \sigma \approx t \sigma$ is derivable, hence $I_E(\gamma)(s) = [s \sigma] = [t \sigma] = I_E(\gamma)(t)$. $\square$

Lemma 4.1.10 ($\Rightarrow^*_E$ is complete). Let $\mathcal{X}$ be a countably infinite set of variables; let $s, t \in T_S(\Sigma, \mathcal{X})$. If $I_E = \forall \bar{x}(s \approx t)$, then $E \Rightarrow^*_E s \approx t$ is derivable.

Proof. Assume that $I_E = \forall \bar{x}(s \approx t)$, i.e., $I_E(\beta)(\forall \bar{x}(s \approx t)) = 1$. Consequently, $I_E(\gamma)(s) = I_E(\gamma)(t)$ for all $\gamma = \beta[x_i \mapsto [t_i] \mid 1 \leq i \leq n]$ with $[t_i] \in \text{sort}(x_i)^{I_E}$. Choose $t_i = x_i$, then $[s] = I_E(\gamma)(s) = I_E(\gamma)(t) = [t]$, so $E \Rightarrow^* s \approx t$ is derivable by definition of $I_E$. $\square$

Theorem 4.1.11 (Birkhoff’s Theorem). Let $\mathcal{X}$ be a countably infinite set of variables, let $E$ be a set of (universally quantified) equations. Then the following properties are equivalent for all $s, t \in T_S(\Sigma, \mathcal{X})$:

1. $s \leftrightarrow^*_E t$.
2. $E \Rightarrow^*_E s \approx t$ is derivable.
3. $s \approx_E t$, i.e., $E = \forall \bar{x}(s \approx t)$.
4. $I_E = \forall \bar{x}(s \approx t)$.
4.2. CRITICAL PAIRS

**Proof.** (1.) ⇔ (2.): Lemma 4.1.5.
(2.) ⇒ (3.): By induction on the size of the derivation for \( E \Rightarrow^* s \approx t \).
(3.) ⇒ (4.): Obvious, since \( I_E = T(\Sigma, \mathcal{X}) / E \) is an \( E \)-algebra.
(4.) ⇒ (2.): Lemma 4.1.10. \( \Box \)

**Universal Algebra**

\( T(\Sigma, \mathcal{X}) / E = T(\Sigma, \mathcal{X}) / \approx_E = T(\Sigma, \mathcal{X}) / \leftrightarrow_E^+ \)

is called the free \( E \)-algebra with generating set \( \mathcal{X} / \approx_E = \{ [x] \mid x \in \mathcal{X} \} \): Every mapping \( \phi : \mathcal{X} / \approx_E \to \mathcal{B} \)

for some \( E \)-algebra \( \mathcal{B} \)

can be extended to a homomorphism \( \phi : T(\Sigma, \mathcal{X}) / E \to \mathcal{B} \).

\( T(\Sigma, \emptyset) / E = T(\Sigma, \emptyset) / \approx_E = T(\Sigma, \emptyset) / \leftrightarrow_E^+ \)

is called the initial \( E \)-algebra.

\( \approx_E = \{ (s, t) \mid E \models s = t \} \)

is called the equational theory of \( E \).

\( \approx_E^\dagger = \{ (s, t) \mid T(\Sigma, \emptyset) / E \models s = t \} \)

is called the inductive theory of \( E \).

**Example 4.1.12.** Let \( E = \{ \forall x(x + 0 \approx x), \forall x\forall y(x + s(y) \approx s(x + y)) \} \). Then \( x + y \approx_E^\dagger y + x \), but \( x + y \not\approx_E y + x \).

---

**4.2 Critical Pairs**

By Theorem 4.1.11 the semantics of \( E \) and \( \leftrightarrow_E^+ \) coincide. In order to decide \( \leftrightarrow_E^+ \) we need to turn \( \rightarrow_E^+ \) in a confluent and terminating relation. If \( \leftrightarrow_E^+ \) is terminating then confluence is equivalent to local confluence, see Newman’s Lemma, Lemma 1.6.6. Local confluence is the following problem for TRS: if \( t_1 \leftarrow_{E} t_0 \rightarrow_{E} t_2 \), does there exist a term \( s \) so that \( t_1 \rightarrow_{E} s \leftarrow_{E} t_2 \)? If the two rewrite steps happen in different subtrees (disjoint redexes) then a repetition of the respective other step yields the common term \( s \). If the two rewrite steps happen below each other (overlap at or below a variable position) again a repetition of the respective other step yields the common term \( s \). If the left-hand sides of the two rules overlap at a non-variable position there is no obvious way to generate \( s \).

More technically two rewrite rules \( l_1 \rightarrow r_1 \) and \( l_2 \rightarrow r_2 \) overlap if there exist some non-variable subterm \( l_1|_p \) such that \( l_2 \) and \( l_1|_p \) have a common instance \( (l_1|_p)[\sigma_1 = \sigma_2] \). If the two rewrite rules do not have common variables, then only a single substitution is necessary, the mgu \( \sigma \) of \( (l_1|_p) \) and \( l_2 \).

**Definition 4.2.1 (Critical Pair).** Let \( l_i \rightarrow r_i \) (\( i = 1, 2 \)) be two rewrite rules in a TRS \( R \) without common variables, i.e., \( \text{vars}(l_1) \cap \text{vars}(l_2) = \emptyset \). Let \( p \in \text{pos}(l_1) \) be a position so that \( l_1|_p \) is not a variable and \( \sigma \) is an mgu of \( l_1|_p \) and \( l_2 \). Then \( r_1 \sigma \leftarrow l_1 \sigma \rightarrow (l_1 \sigma)[r_2 \sigma]|_p \). \( (r_1 \sigma, (l_1 \sigma)[r_2 \sigma]|_p) \) is called a critical pair of \( R \). The critical pair is joinable (or: converges), if \( r_1 \sigma \downarrow_R (l_1 \sigma)[r_2 \sigma]|_p \).

Recall that \( \text{vars}(l_i) \supseteq \text{vars}(r_i) \) for the two rewrite rules by Definition 4.1.1. Furthermore, the definition of the rule includes overlaps of a rule with itself. Such overlaps on top-level are always joinable.
**Theorem 4.2.2** ("Critical Pair Theorem"). A TRS $R$ is locally confluent if and only if all its critical pairs are joinable.

**Proof.** ($\Rightarrow$) Obvious, since joinability of a critical pair is a special case of local confluence.

($\Leftarrow$) Suppose $s$ rewrites to $t_1$ and $t_2$ using rewrite rules $l_i \rightarrow r_i \in R$ at positions $p_i \in \text{pos}(s)$, where $i = 1, 2$. The two rules are variable disjoint, hence $s|_{p_i} = l_i \sigma$ and $t_i = s[r_i \sigma|_p]$. There are two cases to be considered:

1. Either $p_1$ and $p_2$ are in disjoint subtrees ($p_1 \parallel p_2$) or
2. one is a prefix of the other (w.l.o.g., $p_1 \leq p_2$).

Case 1: $p_1 \parallel p_2$. Then $s = s[l_1 \sigma]|_{p_1} [l_2 \sigma]|_{p_2}$, and therefore $t_1 = s[r_1 \sigma]|_{p_1} [r_2 \sigma]|_{p_2}$ and $t_2 = s[l_1 \sigma]|_{p_1} [r_2 \sigma]|_{p_2}$. Let $t_0 = s[r_1 \sigma]|_{p_1} [r_2 \sigma]|_{p_2}$. Then clearly $t_1 \rightarrow_R t_0$ using $l_2 \rightarrow r_2$ and $t_2 \rightarrow_R t_0$ using $l_1 \rightarrow r_1$.

Case 2: $p_1 \leq p_2$.

Case 2.1: $p_2 = p_1 q_1 q_2$, where $l_1|_{q_1}$ is some variable $x$. In other words, the second rewrite step takes place at or below a variable in the first rule. Suppose that $x$ occurs $m$ times in $l_1$ and $n$ times in $r_1$ (where $m \geq 1$ and $n \geq 0$). Then $l_1\rightarrow_R l_1$ by applying $l_2 \rightarrow r_2$ at all positions $p_1 q_1 q_2$, where $q$ is a position of $x$ in $r_1$. Conversely, $t_2 \rightarrow_R t_0$ by applying $l_2 \rightarrow r_2$ at all positions $p_1 q_2$, where $q$ is a position of $x$ in $l_1$ different from $q_1$, and by applying $l_1 \rightarrow r_1$ at $p_1$ with the substitution $\sigma'$, where $\sigma' = \sigma[x \mapsto (x\sigma)|_{r_2 \sigma}|_{q_2}]$.

Case 2.2: $p_2 = p_1 p$, where $p$ is a non-variable position of $l_1$. Then $s|_{p_2} = l_2 \sigma$ and $s|_{p_2} = (s|_{p_1})|_p = (l_1 \sigma)|_p = (l_1|_p)\sigma$, so $\sigma$ is a unifier of $l_2$ and $l_1|_p$. Let $\sigma'$ be the mgu of $l_2$ and $l_1|_p$, then $\sigma = \tau \circ \sigma'$ and $(l_1 \sigma')|(r_2 \sigma')|_p$ is a critical pair. By assumption, it is joinable, so $r_1 \sigma' \rightarrow_R \tau \leftarrow_{\tau} (l_1 \sigma')|(r_2 \sigma')|_p$. Consequently, $t_1 = s[r_1 \sigma]|_{p_1} = s[r_1 \sigma]|_{p_1} \rightarrow_{\tau} s[v\tau]|_{p_1}$ and $t_2 = s[r_2 \sigma]|_{p_2} = s[l_1 \sigma]|_{p_1} = s[l_1 \sigma|(r_2 \sigma')|_p] = s[l_1 \sigma|(r_2 \sigma')|_p] \rightarrow_{\tau} s[v\tau]|_{p_1}$.

Please note that critical pairs between a rule and (a renamed variant of) itself must be considered, except if the overlap is at the root, i.e., $p = \epsilon$, because this critical pair always joins.

**Corollary 4.2.3.** A terminating TRS $R$ is confluent if and only if all its critical pairs are joinable.

**Proof.** By the Theorem 4.2.2 and because every locally confluent and terminating relation $\rightarrow$ is confluent, Newman’s Lemma, Lemma 1.6.6.

**Corollary 4.2.4.** For a finite terminating TRS, confluence is decidable.

**Proof.** For every pair of rules and every non-variable position in the first rule there is at most one critical pair $(u_1, u_2)$. Reduce every $u_i$ to some normal form $u_i'$. If $u'_i = u'_i$ for every critical pair, then $R$ is confluent, otherwise there is some non-confluent situation $u'_1 \rightarrow_R u_1 \leftarrow_R s \rightarrow_R u_2 \rightarrow_R u'_2$. \qed