

Other Inference Systems

- Tableaux
- Instantiation-based methods
 - Resolution-based instance generation
 - Disconnection calculus
 - ...
- Natural deduction
- Sequent calculus/Gentzen calculus
- Hilbert calculus

One major problem with all those calculi concerning automation is that they contain a rule either guessing instances or limiting the use of formulas. So the procedure has to guess instances and/or the number of copies of formulas. For example rules like:

Universal Quantification

$$S \cup \{\forall x \phi\} \Rightarrow S \cup \{\forall x \phi\} \cup \phi\{x \mapsto t\}$$

for some ground term $t \in T_\Sigma$

Existential Quantification

$$S \cup \{\exists x \phi\} \Rightarrow S \cup \{\exists x \phi\} \cup \phi\{x \mapsto a\}$$

for some constant a new to ϕ

4 First-Order Logic with Equality

Equality is the most important relation in mathematics and functional programming.

In principle, problems in first-order logic with equality can be handled by any prover for first-order logic without equality:

4.1 Handling Equality Naively

Proposition 4.1 *Let ϕ be a closed first-order formula with equality. Let $\sim \notin \Pi$ be a new predicate symbol. The set $Eq(\Sigma)$ contains the formulas*

$$\begin{aligned} & \forall x (x \sim x) \\ & \forall x, y (x \sim y \rightarrow y \sim x) \\ & \forall x, y, z (x \sim y \wedge y \sim z \rightarrow x \sim z) \\ & \forall \vec{x}, \vec{y} (x_1 \sim y_1 \wedge \dots \wedge x_n \sim y_n \rightarrow f(x_1, \dots, x_n) \sim f(y_1, \dots, y_n)) \\ & \forall \vec{x}, \vec{y} (x_1 \sim y_1 \wedge \dots \wedge x_m \sim y_m \wedge P(x_1, \dots, x_m) \rightarrow P(y_1, \dots, y_m)) \end{aligned}$$

for every $f \in \Omega$ and $P \in \Pi$. Let $\tilde{\phi}$ be the formula that one obtains from ϕ if every occurrence of \approx is replaced by \sim . Then ϕ is satisfiable if and only if $Eq(\Sigma) \cup \{\tilde{\phi}\}$ is satisfiable.

Proof. Let $\Sigma = (\Omega, \Pi)$, let $\Sigma_1 = (\Omega, \Pi \cup \{\sim\})$.

For the “only if” part assume that ϕ is satisfiable and let \mathcal{A} be a Σ -model of ϕ . Then we define a Σ_1 -algebra \mathcal{B} in such a way that \mathcal{B} and \mathcal{A} have the same universe, $f_{\mathcal{B}} = f_{\mathcal{A}}$ for every $f \in \Omega$, $p_{\mathcal{B}} = p_{\mathcal{A}}$ for every $p \in \Pi$, and $\sim_{\mathcal{B}}$ is the identity relation on the universe. It is easy to check that \mathcal{B} is a model of both $\tilde{\phi}$ and of $Eq(\Sigma)$.

The proof of the “if” part consists of two steps.

Assume that the Σ_1 -algebra $\mathcal{B} = (U_{\mathcal{B}}, (f_{\mathcal{B}} : U^n \rightarrow U)_{f \in \Omega}, (p_{\mathcal{B}} \subseteq U_{\mathcal{B}}^m)_{p \in \Pi \cup \{\sim\}})$ is a model of $Eq(\Sigma) \cup \{\tilde{\phi}\}$. In the first step, we can show that the interpretation $\sim_{\mathcal{B}}$ of \sim in \mathcal{B} is a congruence relation. We will prove this for the symmetry property, the other properties of congruence relations, that is, reflexivity, transitivity, and congruence with respect to functions and predicates are shown analogously. Let $a, a' \in U_{\mathcal{B}}$ such that $a \sim_{\mathcal{B}} a'$. We have to show that $a' \sim_{\mathcal{B}} a$. Since \mathcal{B} is a model of $Eq(\Sigma)$, $\mathcal{B}(\beta)(\forall x, y (x \sim y \rightarrow y \sim x)) = 1$ for every β , hence $\mathcal{B}(\beta[x \mapsto b_1, y \mapsto b_2])(x \sim y \rightarrow y \sim x) = 1$ for every β and every $b_1, b_2 \in U_{\mathcal{B}}$. Set $b_1 = a$ and $b_2 = a'$, then $1 = \mathcal{B}(\beta[x \mapsto a, y \mapsto a'])(x \sim y \rightarrow y \sim x) = (a \sim_{\mathcal{B}} a' \rightarrow a' \sim_{\mathcal{B}} a)$, and since $a \sim_{\mathcal{B}} a'$ holds by assumption, $a' \sim_{\mathcal{B}} a$ must also hold.

In the second step, we will now construct a Σ -algebra \mathcal{A} from \mathcal{B} and the congruence relation $\sim_{\mathcal{B}}$. Let $[a]$ be the congruence class of an element $a \in U_{\mathcal{B}}$ with respect to $\sim_{\mathcal{B}}$. The universe $U_{\mathcal{A}}$ of \mathcal{A} is the set $\{[a] \mid a \in U_{\mathcal{B}}\}$ of congruence classes of the universe of \mathcal{B} . For a function symbol $f \in \Omega$, we define $f_{\mathcal{A}}([a_1], \dots, [a_n]) = [f_{\mathcal{B}}(a_1, \dots, a_n)]$, and for a predicate symbol $p \in \Pi$, we define $([a_1], \dots, [a_n]) \in p_{\mathcal{A}}$ if and only if $(a_1, \dots, a_n) \in p_{\mathcal{B}}$. Observe that this is well-defined: If we take different representatives of the same congruence class, we get the same result by congruence of $\sim_{\mathcal{B}}$. Now for every Σ -term t and every \mathcal{B} -assignment β , $[\mathcal{B}(\beta)(t)] = \mathcal{A}(\gamma)(t)$, where γ is the \mathcal{A} -assignment that maps every variable x to $[\beta(x)]$, and analogously for every Σ -formula ψ , $\mathcal{B}(\beta)(\tilde{\psi}) = \mathcal{A}(\gamma)(\psi)$. Both properties can easily be shown by structural induction. Consequently, \mathcal{A} is a model of ϕ . \square

By giving the equality axioms explicitly, first-order problems with equality can in principle be solved by *FSTP*.

But this is unfortunately not efficient, mainly due to the transitivity axiom.

Equality is theoretically difficult: First-order functional programming is Turing-complete.

But: *FSTP* cannot even solve equational problems that are intuitively easy.

Consequence: to handle equality efficiently, knowledge must be integrated into the theorem prover.

Roadmap

How to proceed:

Term rewrite systems
Expressing semantic consequence syntactically
Knuth-Bendix-Completion
Entailment for equations
(Superposition for first-order clauses with equality)

4.2 Term Rewrite Systems

Let E be a set of (implicitly universally quantified) equations.

The *rewrite relation* $\rightarrow_E \subseteq T_\Sigma(X) \times T_\Sigma(X)$ is defined by

$s \rightarrow_E t$ iff there exist $(l \approx r) \in E$, $p \in \text{pos}(s)$,
and $\sigma : X \rightarrow T_\Sigma(X)$,
such that $s|_p = l\sigma$ and $t = s[r\sigma]_p$.

An instance of the lhs (left-hand side) of an equation is called a *redex* (reducible expression). *Contracting* a redex means replacing it with the corresponding instance of the rhs (right-hand side) of the rule.

An equation $l \approx r$ is also called a *rewrite rule*, if l is not a variable and $\text{vars}(l) \supseteq \text{vars}(r)$.

Notation: $l \rightarrow r$.

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

We say that a set of equations E or a TRS R is *terminating*, if the rewrite relation \rightarrow_E or \rightarrow_R has this property.

(Analogously for other properties of (abstract) rewrite systems).

Note: If E is terminating, then it is a TRS.

Rewrite Relations

Corollary 4.2 *If E is convergent (i. e., terminating and confluent), then $s \approx_E t$ if and only if $s \leftrightarrow_E^* t$ if and only if $s \downarrow_E = t \downarrow_E$.*

Corollary 4.3 *If E is finite and convergent, then \approx_E is decidable.*

Reminder:

If E is terminating, then it is confluent if and only if it is locally confluent.

Problems:

Show local confluence of E .

Show termination of E .

Transform E into an equivalent set of equations that is locally confluent and terminating.

E-Algebras

Let E be a set of universally quantified equations. A model of E is also called an E -algebra.

If $E \models \forall \vec{x}(s \approx t)$, i. e., $\forall \vec{x}(s \approx t)$ is valid in all E -algebras, we write this also as $s \approx_E t$.

Goal:

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 equations over $T_\Sigma(X)$. The following inference system allows to derive consequences of E :

$$\mathcal{I} \frac{}{t \approx t} \quad (\text{Reflexivity})$$

$$\mathcal{I} \frac{t \approx t'}{t' \approx t} \quad (\text{Symmetry})$$

$$\mathcal{I} \frac{t \approx t' \quad t' \approx t''}{t \approx t''} \quad (\text{Transitivity})$$

$$\mathcal{I} \frac{t_1 \approx t'_1 \quad \dots \quad t_n \approx t'_n}{f(t_1, \dots, t_n) \approx f(t'_1, \dots, t'_n)} \quad \text{for any } f/n \quad (\text{Congruence})$$

$$\mathcal{I} \frac{t \approx t'}{t\sigma \approx t'\sigma} \quad \text{for any substitution } \sigma \quad (\text{Instance})$$

Lemma 4.4 *The following properties are equivalent:*

- (i) $s \leftrightarrow_E^* t$
- (ii) $E \Rightarrow^* s \approx t$.

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

Recall that the before inference rules of the form $\mathcal{I} \frac{A_1 \dots A_k}{B}$ are abbreviations for rewrite rules $E \uplus \{A_1, \dots, A_k\} \Rightarrow E \cup \{A_1, \dots, A_k, B\}$.

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

Constructing a *quotient algebra*:

Let X be a set of variables.

For $t \in \mathsf{T}_\Sigma(X)$ let $[t] = \{t' \in \mathsf{T}_\Sigma(X) \mid E \Rightarrow^* t \approx t'\}$ be the *congruence class* of t .

Define a Σ -algebra $\mathsf{T}_\Sigma(X)/E$ (abbreviated by \mathcal{T}) as follows:

$$U_{\mathcal{T}} = \{[t] \mid t \in \mathsf{T}_\Sigma(X)\},$$

$$f_{\mathcal{T}}([t_1], \dots, [t_n]) = [f(t_1, \dots, t_n)] \text{ for } f \in \Omega.$$

Lemma 4.5 $f_{\mathcal{T}}$ is well-defined: If $[t_i] = [t'_i]$, then $[f(t_1, \dots, t_n)] = [f(t'_1, \dots, t'_n)]$.

Proof. Follows directly from the *Congruence* rule for \Rightarrow^* . □

Lemma 4.6 $\mathcal{T} = \mathsf{T}_\Sigma(X)/E$ is an E -algebra.

Proof. Let $\forall x_1 \dots x_n (s \approx t)$ be an equation in E ; let β be an arbitrary assignment.

We have to show that $\mathcal{T}(\beta)(\forall \vec{x}(s \approx t)) = 1$, or equivalently, that $\mathcal{T}(\gamma)(s) = \mathcal{T}(\gamma)(t)$ for all $\gamma = \beta[x_i \mapsto [t_i] \mid 1 \leq i \leq n]$ with $[t_i] \in U_{\mathcal{T}}$.

Let $\sigma = \{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}$, then $s\sigma \in \mathcal{T}(\gamma)(s)$ and $t\sigma \in \mathcal{T}(\gamma)(t)$.

By the *Instance* rule, $E \Rightarrow^* s\sigma \approx t\sigma$, hence $\mathcal{T}(\gamma)(s) = [s\sigma] = [t\sigma] = \mathcal{T}(\gamma)(t)$. □

Lemma 4.7 Let X be a countably infinite set of variables; let $s, t \in T_\Sigma(X)$. If $T_\Sigma(X)/E \models \forall \vec{x}(s \approx t)$, then $E \Rightarrow^* s \approx t$.

Proof. Assume that $\mathcal{T} \models \forall \vec{x}(s \approx t)$, i. e., $\mathcal{T}(\beta)(\forall \vec{x}(s \approx t)) = 1$. Consequently, $\mathcal{T}(\gamma)(s) = \mathcal{T}(\gamma)(t)$ for all $\gamma = \beta[x_i \mapsto [t_i] \mid 1 \leq i \leq n]$ with $[t_i] \in U_{\mathcal{T}}$.

Choose $t_i = x_i$, then $[s] = \mathcal{T}(\gamma)(s) = \mathcal{T}(\gamma)(t) = [t]$, so $E \Rightarrow^* s \approx t$ by definition of \mathcal{T} . \square

Theorem 4.8 (“Birkhoff’s Theorem”) Let 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 $s, t \in T_\Sigma(X)$:

- (i) $s \leftrightarrow_E^* t$.
- (ii) $E \Rightarrow^* s \approx t$.
- (iii) $s \approx_E t$, i. e., $E \models \forall \vec{x}(s \approx t)$.
- (iv) $T_\Sigma(X)/E \models \forall \vec{x}(s \approx t)$.

Proof. (i) \Leftrightarrow (ii): Lemma 4.4.

(ii) \Rightarrow (iii): By induction on the size of the derivation for $E \Rightarrow^* s \approx t$.

(iii) \Rightarrow (iv): Obvious, since $\mathcal{T} = T_\Sigma(X)/E$ is an E -algebra.

(iv) \Rightarrow (ii): Lemma 4.7. \square

Universal Algebra

$T_\Sigma(X)/E = T_\Sigma(X)/\approx_E = T_\Sigma(X)/\leftrightarrow_E^*$ is called the *free E -algebra* with generating set $X/\approx_E = \{[x] \mid x \in X\}$:

Every mapping $\varphi : X/\approx_E \rightarrow \mathcal{B}$ for some E -algebra \mathcal{B} can be extended to a homomorphism $\hat{\varphi} : T_\Sigma(X)/E \rightarrow \mathcal{B}$.

$T_\Sigma(\emptyset)/E = T_\Sigma(\emptyset)/\approx_E = T_\Sigma(\emptyset)/\leftrightarrow_E^*$ is called the *initial E -algebra*.

$\approx_E = \{(s, t) \mid E \models s \approx t\}$ is called the *equational theory* of E .

$\approx_E^I = \{(s, t) \mid T_\Sigma(\emptyset)/E \models s \approx t\}$ is called the *inductive theory* of E .

Example:

Let $E = \{\forall x(x + 0 \approx x), \forall x \forall y(x + s(y) \approx s(x + y))\}$. Then $x + y \approx_E^I y + x$, but $x + y \not\approx_E y + x$.

4.3 Critical Pairs

Showing local confluence (Sketch):

Problem: If $t_1 \xrightarrow{E} t_0 \xrightarrow{E} t_2$, does there exist a term s such that $t_1 \xrightarrow{E}^* s \xrightarrow{E}^* t_2$?

If the two rewrite steps happen in different subtrees (disjoint redexes): yes.

If the two rewrite steps happen below each other (overlap at or below a variable position): yes.

If the left-hand sides of the two rules overlap at a non-variable position: needs further investigation.

Question:

Are there rewrite rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ such that some subterm $l_1|_p$ and l_2 have a common instance $(l_1|_p)\sigma_1 = l_2\sigma_2$?

Observation:

If we assume w.o.l.o.g. that the two rewrite rules do not have common variables, then only a single substitution is necessary: $(l_1|_p)\sigma = l_2\sigma$.

Further observation:

The mgu of $l_1|_p$ and l_2 subsumes all unifiers σ of $l_1|_p$ and l_2 .

Let $l_i \rightarrow r_i$ ($i = 1, 2$) be two rewrite rules in a TRS R whose variables have been renamed such that $\text{vars}(l_1) \cap \text{vars}(l_2) = \emptyset$. (Remember that $\text{vars}(l_i) \supseteq \text{vars}(r_i)$.)

Let $p \in \text{pos}(l_1)$ be a position such that $l_1|_p$ is not a variable and σ is an mgu of $l_1|_p$ and l_2 .

Then $r_1\sigma \leftarrow l_1\sigma \rightarrow (l_1\sigma)[r_2\sigma]_p$.

$\langle r_1\sigma, (l_1\sigma)[r_2\sigma]_p \rangle$ is called a *critical pair* of R .

The critical pair is *joinable* (or: converges), if $r_1\sigma \downarrow_R (l_1\sigma)[r_2\sigma]_p$.

Theorem 4.9 (“Critical Pair Theorem”) *A TRS R is locally confluent if and only if all its critical pairs are joinable.*

Proof. “only if”: obvious, since joinability of a critical pair is a special case of local confluence.

“if”: 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 $i = 1, 2$. Without loss of generality, we can assume that the two rules are variable disjoint, hence $s|_{p_i} = l_i\theta$ and $t_i = s[r_i\theta]_{p_i}$.

We distinguish between two cases: Either p_1 and p_2 are in disjoint subtrees ($p_1 \parallel p_2$), or one is a prefix of the other (w.o.l.o.g., $p_1 \leq p_2$).

Case 1: $p_1 \parallel p_2$.

Then $s = s[l_1\theta]_{p_1}[l_2\theta]_{p_2}$, and therefore $t_1 = s[r_1\theta]_{p_1}[l_2\theta]_{p_2}$ and $t_2 = s[l_1\theta]_{p_1}[r_2\theta]_{p_2}$.

Let $t_0 = s[r_1\theta]_{p_1}[r_2\theta]_{p_2}$. 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: $p_2 = p_1q_1q_2$, 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 $t_1 \rightarrow_R^* t_0$ by applying $l_2 \rightarrow r_2$ at all positions $p_1q'q_2$, 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_1qq_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 θ' , where $\theta' = \theta[x \mapsto (x\theta)[r_2\theta]_{q_2}]$.

Case 2.2: $p_2 = p_1p$, where p is a non-variable position of l_1 .

Then $s|_{p_2} = l_2\theta$ and $s|_{p_2} = (s|_{p_1})|_p = (l_1\theta)|_p = (l_1|_p)\theta$, so θ is a unifier of l_2 and $l_1|_p$.

Let σ be the mgu of l_2 and $l_1|_p$, then $\theta = \tau \circ \sigma$ and $\langle r_1\sigma, (l_1\sigma)[r_2\sigma]_p \rangle$ is a critical pair.

By assumption, it is joinable, so $r_1\sigma \rightarrow_R^* v \leftarrow_R^* (l_1\sigma)[r_2\sigma]_p$.

Consequently, $t_1 = s[r_1\theta]_{p_1} = s[r_1\sigma\tau]_{p_1} \rightarrow_R^* s[v\tau]_{p_1}$ and $t_2 = s[r_2\theta]_{p_2} = s[(l_1\theta)[r_2\theta]_p]_{p_1} = s[(l_1\sigma\tau)[r_2\sigma\tau]_p]_{p_1} = s[(l_1\sigma)[r_2\sigma]_p\tau]_{p_1} \rightarrow_R^* s[v\tau]_{p_1}$.

This completes the proof of the Critical Pair Theorem. \square

Note: Critical pairs between a rule and (a renamed variant of) itself must be considered – except if the overlap is at the root (i. e., $p = \varepsilon$).

Corollary 4.10 *A terminating TRS R is confluent if and only if all its critical pairs are joinable.*

Proof. By Newman's Lemma and the Critical Pair Theorem. \square

Corollary 4.11 *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 $\langle u_1, u_2 \rangle$.

Reduce every u_i to some normal form u'_i . If $u'_1 = u'_2$ for every critical pair, then R is confluent, otherwise there is some non-confluent situation $u'_1 \xleftarrow{*}_R u_1 \leftarrow_R s \rightarrow_R u_2 \rightarrow_R^* u'_2$. \square