Knuth-Bendix Completion: Correctness Proof

If we run the completion procedure on a set E of equations, different things can happen:

(1) We reach a state where no more inference rules are applicable and E is not empty. $\Rightarrow$ Failure (try again with another ordering?)

(2) We reach a state where E is empty and all critical pairs between the rules in the current R have been checked.

(3) The procedure runs forever.

In order to treat these cases simultaneously, we need some definitions.

A (finite or infinite sequence) $E_0, R_0 \vdash E_1, R_1 \vdash E_2, R_2 \vdash \ldots$ with $R_0 = \emptyset$ is called a *run* of the completion procedure with input E_0 and $\succ$.

For a run, $E_\infty = \bigcup_{i\geq 0} E_i$ and $R_\infty = \bigcup_{i\geq 0} R_i$.

The sets of *persistent equations or rules* of the run are $E_* = \bigcup_{i\geq 0} \bigcap_{j\geq i} E_j$ and $R_* = \bigcup_{i\geq 0} \bigcap_{j\geq i} R_j$.

Note: If the run is finite and ends with E_n, R_n, then $E_* = E_n$ and $R_* = R_n$.

A run is called *fair*, if $CP(R_*) \subseteq E_\infty$ (i. e., if every critical pair between persisting rules is computed at some step of the derivation).

Goal:

Show: If a run is fair and E_* is empty, then R_* is convergent and equivalent to E_0.

In particular: If a run is fair and E_* is empty, then $\approx_{E_0} = \approx_{E_\infty \cup R_\infty} = \leftrightarrow^*_{E_\infty \cup R_\infty} = \downarrow_{R_*}$.

General assumptions from now on:

$E_0, R_0 \vdash E_1, R_1 \vdash E_2, R_2 \vdash \ldots$ is a fair run.

R_0 and E_* are empty.

A *proof* of $s \approx t$ in $E_\infty \cup R_\infty$ is a finite sequence $(s_0, \ldots, s_n)$ such that $s = s_0$, $t = s_n$, and for all $i \in \{1, \ldots, n\}$:

(1) $s_{i-1} \leftrightarrow_{E_\infty} s_i$, or

(2) $s_{i-1} \rightarrow_{R_\infty} s_i$, or

(3) $s_{i-1} \leftarrow_{R_\infty} s_i$.

The pairs (s_{i-1}, s_i) are called *proof steps*.

A proof is called a *rewrite proof in* R_*, if there is a $k \in \{0, \ldots, n\}$ such that $s_{i-1} \rightarrow_{R_*} s_i$ for $1 \leq i \leq k$ and $s_{i-1} \leftarrow_{R_*} s_i$ for $k+1 \leq i \leq n$

Idea (Bachmair, Dershowitz, Hsiang):

Define a well-founded ordering on proofs, such that for every proof that is not a rewrite proof in R_* there is an equivalent smaller proof.

Consequence: For every proof there is an equivalent rewrite proof in R_*.

We associate a *cost* $c(s_{i-1}, s_i)$ with every proof step as follows:

(1) If $s_{i-1} \leftrightarrow_{E_\infty} s_i$, then $c(s_{i-1}, s_i) = (\{s_{i-1}, s_i\}, -, -)$, where the first component is a multiset of terms and $-$ denotes an arbitrary (irrelevant) term.

(2) If $s_{i-1} \rightarrow_{R_\infty} s_i$ using $l \rightarrow r$, then $c(s_{i-1}, s_i) = (\{s_{i-1}\}, l, s_i)$.

(3) If $s_{i-1} \leftarrow_{R_\infty} s_i$ using $l \rightarrow r$, then $c(s_{i-1}, s_i) = (\{s_i\}, l, s_{i-1})$.

Proof steps are compared using the lexicographic combination of the multiset extension of the reduction ordering $\succ$, the encompassment ordering $\sqsupset$, and the reduction ordering $\succ$.

The cost $c(P)$ of a proof P is the multiset of the costs of its proof steps.

The *proof ordering* $\succ_C$ compares the costs of proofs using the multiset extension of the proof step ordering.

Lemma 4.41 *$\succ_C$ is a well-founded ordering.*

Lemma 4.42 *Let P be a proof in $E_\infty \cup R_\infty$. If P is not a rewrite proof in R_*, then there exists an equivalent proof P' in $E_\infty \cup R_\infty$ such that $P \succ_C P'$.*

Proof. If P is not a rewrite proof in R_*, then it contains

(a) a proof step that is in E_∞, or
(b) a proof step that is in $R_\infty \setminus R_*$, or
(c) a subproof $s_{i-1} \leftarrow_{R_*} s_i \rightarrow_{R_*} s_{i+1}$ (peak).

We show that in all three cases the proof step or subproof can be replaced by a smaller subproof:

Case (a): A proof step using an equation $s \doteq t$ is in E_∞. This equation must be deleted during the run.

If $s \doteq t$ is deleted using *Orient*:

$$\ldots s_{i-1} \leftrightarrow_{E_\infty} s_i \ldots \quad \Longrightarrow \quad \ldots s_{i-1} \rightarrow_{R_\infty} s_i \ldots$$

If $s \doteq t$ is deleted using *Delete*:

$$\ldots s_{i-1} \leftrightarrow_{E_\infty} s_{i-1} \ldots \quad \Longrightarrow \quad \ldots s_{i-1} \ldots$$

If $s \doteq t$ is deleted using *Simplify-Eq*:

$$\ldots s_{i-1} \leftrightarrow_{E_\infty} s_i \ldots \quad \Longrightarrow \quad \ldots s_{i-1} \rightarrow_{R_\infty} s' \leftrightarrow_{E_\infty} s_i \ldots$$

Case (b): A proof step using a rule $s \rightarrow t$ is in $R_\infty \setminus R_*$. This rule must be deleted during the run.

If $s \rightarrow t$ is deleted using *R-Simplify-Rule*:

$$\ldots s_{i-1} \rightarrow_{R_\infty} s_i \ldots \quad \Longrightarrow \quad \ldots s_{i-1} \rightarrow_{R_\infty} s' \leftarrow_{R_\infty} s_i \ldots$$

If $s \rightarrow t$ is deleted using *L-Simplify-Rule*:

$$\ldots s_{i-1} \rightarrow_{R_\infty} s_i \ldots \quad \Longrightarrow \quad \ldots s_{i-1} \rightarrow_{R_\infty} s' \leftrightarrow_{E_\infty} s_i \ldots$$

Case (c): A subproof has the form $s_{i-1} \leftarrow_{R_*} s_i \rightarrow_{R_*} s_{i+1}$.

If there is no overlap or a non-critical overlap:

$$\ldots s_{i-1} \leftarrow_{R_*} s_i \rightarrow_{R_*} s_{i+1} \ldots \Longrightarrow \ldots s_{i-1} \rightarrow^*_{R_*} s' \leftarrow^*_{R_*} s_{i+1} \ldots$$

If there is a critical pair that has been added using *Deduce*:

$$\ldots s_{i-1} \leftarrow_{R_*} s_i \rightarrow_{R_*} s_{i+1} \ldots \Longrightarrow \quad \ldots s_{i-1} \leftrightarrow_{E_\infty} s_{i+1} \ldots$$

In all cases, checking that the replacement subproof is smaller than the replaced subproof is routine. □

Theorem 4.43 *Let $E_0, R_0 \vdash E_1, R_1 \vdash E_2, R_2 \vdash \ldots$ be a fair run and let R_0 and E_* be empty. Then*

(1) every proof in $E_\infty \cup R_\infty$ is equivalent to a rewrite proof in R_*,

(2) R_* is equivalent to E_0, and

(3) R_* is convergent.

Proof. (1) By well-founded induction on $\succ_C$ using the previous lemma.

(2) Clearly $\approx_{E_\infty \cup R_\infty} = \approx_{E_0}$. Since $R_* \subseteq R_\infty$, we get $\approx_{R_*} \subseteq \approx_{E_\infty \cup R_\infty}$. On the other hand, by (1), $\approx_{E_\infty \cup R_\infty} \subseteq \approx_{R_*}$.

(3) Since $\rightarrow_{R_*} \subseteq \succ$, R_* is terminating. By (1), R_* is confluent. $\square$

Knuth-Bendix Completion: Outlook

Classical completion:

Tries to transform a set E of equations into an equivalent convergent term rewrite system.

Fails, if an equation can neither be oriented nor deleted.

Unfailing completion:

Use an ordering $\succ$ that is total on ground terms.

If an equation cannot be oriented, use it in both directions for rewriting (except if that would yield a larger term). In other words, consider the relation $\leftrightarrow_E \cap \not\prec$.

Special case of superposition (see next chapter).

4.7 Superposition

Goal:

Combine the ideas of ordered resolution (overlap maximal literals in a clause) and Knuth-Bendix completion (overlap maximal sides of equations) to get a calculus for equational clauses.

Recapitulation: Equational Clauses

Atom: either $P(s_1, \dots, s_m)$ with $P \in \Pi$ or $s \approx t$.

Literal: Atom or negated atom.

Clause: (possibly empty) disjunction of literals (all variables implicitly universally quantified).

For refutational theorem proving, it is sufficient to consider sets of clauses: every first-order formula F can be translated into a set of clauses N such that F is unsatisfiable if and only if N is unsatisfiable.

In the non-equational case, unsatisfiability can for instance be checked using the (ordered) resolution calculus.

Recapitulation: Ordered Resolution

(Ordered) resolution: inference rules:

	Ground case:	Non-ground case:
Resolution:	$\dfrac{D' \vee A \qquad C' \vee \neg A}{D' \vee C'}$	$\dfrac{D' \vee A \qquad C' \vee \neg A'}{(D' \vee C')\sigma}$ where $\sigma = \mathrm{mgu}(A, A')$.
Factoring:	$\dfrac{C' \vee A \vee A}{C' \vee A}$	$\dfrac{C' \vee A \vee A'}{(C' \vee A)\sigma}$ where $\sigma = \mathrm{mgu}(A, A')$.

Ordering restrictions:

Let $\succ$ be a well-founded and total ordering on ground atoms.

Literal ordering $\succ_L$: compares literals by comparing lexicographically first the respective atoms using $\succ$ and then their polarities (negative $>$ positive).

Clause ordering $\succ_C$: compares clauses by comparing their multisets of literals using the multiset extension of $\succ_L$.

Ordering restrictions (ground case):

Inference are necessary only if the following conditions are satisfied:

- The left premise of a Resolution inference is not larger than or equal to the right premise.
- The literals that are involved in the inferences ($[\neg]\ A$) are maximal in the respective clauses (strictly maximal for the left premise of Resolution).

Ordering restrictions (non-ground case):

Lift the ground ordering to non-ground literals: A literal L is called [strictly] maximal in a clause C if and only if there exists a ground substitution σ such that for all other literals L' in C: $L\sigma \not\prec L'\sigma$ [$L\sigma \not\preceq L'\sigma$].

Recapitulation: Refutational Completeness

Resolution is (even with ordering restrictions) refutationally complete:

Dynamic view of refutational completeness:

If N is unsatisfiable ($N \models \bot$) then *fair* derivations from N produce $\bot$.

Static view of refutational completeness:

If N is *saturated*, then N is unsatisfiable if and only if $\bot \in N$.

Proving refutational completeness for the ground case:

We have to show:

If N is saturated (i. e., if sufficiently many inferences have been computed), and $\bot \notin N$, then N is satisfiable (i. e., has a model).

Model construction:

Suppose that N be saturated and $\bot \notin N$. We inspect all clauses in N in ascending order and construct a sequence of Herbrand interpretations (starting with the empty interpretation: all atoms are false).

If a clause C is false in the current interpretation, and has a positive and strictly maximal literal A, then extend the current interpretation such that C becomes true: add A to the current interpretation. (Then C is called *productive*.)

Otherwise, leave the current interpretation unchanged.

The sequence of interpretations has the following properties:

(1) If an atom is true in some interpretation, then it remains true in all future interpretations.

(2) If a clause is true at the time where it is inspected, then it remains true in all future interpretations.

(3) If a clause $C = C' \vee A$ is productive, then C remains true and C' remains false in all future interpretations.

Show by induction: if N is saturated and $\bot \notin N$, then every clause in N is either true at the time where it is inspected or productive.

Note:
For the induction proof, it is not necessary that the conclusion of an inference is contained in N. It is sufficient that it is redundant w. r. t. N.

N is called *saturated up to redundancy* if the conclusion of every inference from clauses in $N \setminus \mathit{Red}(N)$ is contained in $N \cup \mathit{Red}(N)$.

Proving refutational completeness for the non-ground case:

If $C_i\theta$ is a ground instance of the clause C_i for $i \in \{0, \dots, n\}$ and

$$\frac{C_n, \dots, C_1}{C_0}$$

and

$$\frac{C_n\theta, \dots, C_1\theta}{C_0\theta}$$

are inferences, then the latter inference is called a *ground instance* of the former.

For a set N of clauses, let $G_\Sigma(N)$ be the set of all ground instances of clauses in N.

Construct the interpretation from the set $G_\Sigma(N)$ of all ground instances of clauses in N:

N is saturated and does not contain $\bot$
$\Rightarrow$ $G_\Sigma(N)$ is saturated and does not contain $\bot$
$\Rightarrow$ $G_\Sigma(N)$ has a Herbrand model I
$\Rightarrow$ I is a model of N.

Observation

It is possible to encode an arbitrary predicate p using a function f_p and a new constant tt:

$$\begin{array}{rcr} P(t_1,\dots,t_n) & \rightsquigarrow & f_P(t_1,\dots,t_n) \approx tt \\ \neg\, P(t_1,\dots,t_n) & \rightsquigarrow & \neg\, f_P(t_1,\dots,t_n) \approx tt \end{array}$$

In equational logic it is therefore sufficient to consider the case that $\Pi = \emptyset$, i. e., equality is the only predicate symbol.

Abbreviation: $s \not\approx t$ instead of $\neg\, s \approx t$.

The Superposition Calculus – Informally

Conventions:

From now on: $\Pi = \emptyset$ (equality is the only predicate).

Inference rules are to be read modulo symmetry of the equality symbol.

We will first explain the ideas and motivations behind the superposition calculus and its completeness proof. Precise definitions will be given later.

Ground inference rules:

Pos. Superposition: $$\frac{D' \lor t \approx t' \qquad C' \lor s[t] \approx s'}{D' \lor C' \lor s[t'] \approx s'}$$

Neg. Superposition: $$\frac{D' \lor t \approx t' \qquad C' \lor s[t] \not\approx s'}{D' \lor C' \lor s[t'] \not\approx s'}$$

Equality Resolution: $$\frac{C' \lor s \not\approx s}{C'}$$

(Note: We will need one further inference rule.)

Ordering restrictions:

Some considerations:

The literal ordering must depend primarily on the larger term of an equation.

As in the resolution case, negative literals must be a bit larger than the corresponding positive literals.

Additionally, we need the following property: If $s \succ t \succ u$, then $s \not\approx u$ must be larger than $s \approx t$. In other words, we must compare first the larger term, then the polarity, and finally the smaller term.

The following construction has the required properties:

Let $\succ$ be a *reduction ordering that is total on ground terms.*

To a positive literal $s \approx t$, we assign the multiset $\{s, t\}$, to a negative literal $s \not\approx t$ the multiset $\{s, s, t, t\}$. The *literal ordering* $\succ_L$ compares these multisets using the multiset extension of $\succ$.

The *clause ordering* $\succ_C$ compares clauses by comparing their multisets of literals using the multiset extension of $\succ_L$.

Ordering restrictions:

Ground inferences are necessary only if the following conditions are satisfied:

- In superposition inferences, the left premise is smaller than the right premise.
- The literals that are involved in the inferences are maximal in the respective clauses (strictly maximal for positive literals in superposition inferences).
- In these literals, the lhs is greater than or equal to the rhs (in superposition inferences: greater than the rhs).

Model construction:

We want to use roughly the same ideas as in the completeness proof for resolution.

But: a Herbrand interpretation does not work for equality: The equality symbol $\approx$ must be interpreted by equality in the interpretation.

Solution: Define a set E of ground equations and take $\mathrm{T}_\Sigma(\emptyset)/E = \mathrm{T}_\Sigma(\emptyset)/{\approx_E}$ as the universe.

Then two ground terms s and t are equal in the interpretation, if and only if $s \approx_E t$.

If E is a terminating and confluent rewrite system R, then two ground terms s and t are equal in the interpretation, if and only if $s \downarrow_R t$.

One problem:

In the completeness proof for the resolution calculus, the following property holds:

If $C = C' \vee A$ with a strictly maximal and positive literal A is false in the current interpretation, then adding A to the current interpretation cannot make any literal of C' true.