Completeness of Superposition

Andrei Voronkov

The University of Manchester
EasyChair

Bag Extension of an Ordering

Bag = finite multiset.
Let $>$ be any ordering on a set X. The bag extension of $>$ is a binary relation $>^{bag}$, on bags over X, defined as the smallest transitive relation on bags such that

$$\{x, y_1, \ldots, y_n\} >^{bag} \{x_1, \ldots, x_m, y_1, \ldots, y_n\}$$
$$\text{if } x > x_i \text{ for all } i \in \{1 \ldots m\},$$

where $m \geq 0$.
Idea: a bag becomes smaller if we replace an element by any finite number of smaller elements.
The following results are known about the bag extensions of orderings:

1. $>^{bag}$ is an ordering;
2. If $>$ is total, then so is $>^{bag}$;
3. If $>$ is well-founded, then so is $>^{bag}$.

Clause Orderings

From now on consider clauses also as bags of literals. Note:

- we have an ordering $\succ$ for comparing literals;
- a clause is a bag of literals.

Hence

- we can compare clauses using the bag extension $\succ^{bag}$ of $\succ$.

For simplicity we denote the multiset ordering also by $\succ$.

Redundancy

A clause C is called redundant w.r.t. S if it is a logical consequence of clauses in S strictly smaller than C.

If C is redundant w.r.t. S and $C \in S$, we say that C is redundant in S.

Inference Process with Redundancy

Let $\mathbb{I}$ be an inference system. Consider an inference process with two kinds of step $S_i \Rightarrow S_{i+1}$:

1. Adding the conclusion of an $\mathbb{I}$-inference with premises in S_i.
2. Deletion of a clause redundant in S_i, that is

$$S_{i+1} = S_i - \{C\},$$

   where C is redundant in S_i.

Fairness: Persistent Clauses and Limit

Consider an inference process

$$S_0 \Rightarrow S_1 \Rightarrow S_2 \Rightarrow \ldots$$

A clause C is called persistent if

$$\exists i \forall j \geq i (C \in S_j).$$

The limit S_ω of the inference process is the set of all persistent clauses:

$$S_\omega = \bigcup_{i=0,1,\ldots} \bigcap_{j \geq i} S_j.$$

Fairness

The process is called $\mathbb{I}$-fair if for every $\mathbb{I}$-inference

$$\frac{C_1 \quad \ldots \quad C_n}{C} \ ,$$

if $\{C_1, \ldots, C_n\} \subseteq S_\omega$, then $C \in S_i$ for some i.

That is, every inference with persistent premises must be applied at some step, unless its conclusion was already occurring in some search space S_i.

Ground Superposition Inference System $\mathbb{S}\mathrm{up}_{\succ,\sigma}$

Superposition: (right and left)

$$\frac{\underline{l \simeq r} \vee C \quad \underline{s[l] \simeq t} \vee D}{s[r] \simeq t \vee C \vee D}\ (\mathrm{Sup}), \quad \frac{\underline{l \simeq r} \vee C \quad \underline{s[l] \not\simeq t} \vee D}{s[r] \not\simeq t \vee C \vee D}\ (\mathrm{Sup}),$$

where (i) $l \succ r$;
(ii) l is greater than any term in C
(iii) $s[l] \succ t$;

Equality Resolution:

$$\frac{\underline{s \not\simeq s} \vee C}{C}\ (\mathrm{ER}),$$

Equality Factoring:

$$\frac{\underline{l \simeq r} \vee l \simeq r' \vee C}{l \simeq r \vee r \not\simeq r' \vee C}\ (\mathrm{EF}),$$

where (i) $l \succ r$;
(ii) $l \simeq r$ is a maximal literal in $l \simeq r \vee l \simeq r' \vee C$.

Ground Superposition Inference System $\mathbb{S}\mathrm{up}_{\succ,\sigma}$

Superposition: (right and left)

$$\frac{\underline{l \simeq r} \vee C \quad \underline{s[l] \simeq t} \vee D}{s[r] \simeq t \vee C \vee D} \text{ (Sup)}, \quad \frac{\underline{l \simeq r} \vee C \quad \underline{s[l] \not\simeq t} \vee D}{s[r] \not\simeq t \vee C \vee D} \text{ (Sup)},$$

where (i) $l \succ r$;
(ii) l is greater than any term in C
(iii) $s[l] \succ t$;
(iv) $s[l] \simeq t$ is the greatest literal in $s[l] \simeq t \vee D$ (only for $\simeq$)

Equality Resolution:

$$\frac{\underline{s \not\simeq s} \vee C}{C} \text{ (ER)},$$

Equality Factoring:

$$\frac{\underline{l \simeq r} \vee l \simeq r' \vee C}{l \simeq r \vee r \not\simeq r' \vee C} \text{ (EF)},$$

where (i) $l \succ r$;
(ii) $l \simeq r$ is a maximal literal in $l \simeq r \vee l \simeq r' \vee C$.

Completeness of $\mathbb{S}\mathrm{up}_{\succ,\sigma}$

Completeness Theorem. Let $\succ$ be a simplification ordering and σ a well-behaving selection function. Let also

1. S_0 be a set of clauses;
2. $S_0 \Rightarrow S_1 \Rightarrow S_2 \Rightarrow \ldots$ be a fair $\mathbb{S}\mathrm{up}_{\succ,\sigma}$-inference process.

Then S_0 is unsatisfiable if and only if $\square \in S_i$ for some i.

Saturation up to Redundancy

A set S of clauses is called saturated up to redundancy if for every $\mathbb{I}$-inference

$$\frac{C_1 \quad \ldots \quad C_n}{C}$$

with premises in S, either

1. $C \in S$; or
2. C is redundant w.r.t. S, that is, $S_{\prec C} \models C$.

Proof of Completeness

A trace of a clause C: a set of clauses $\{C_1, \ldots, C_n\} \subseteq S_\omega$ such that

1. $C \succeq C_i$ for all $i = 1, \ldots, n$;
2. $C_1, \ldots, C_n \models C$.

Lemma 1. Every clause C occurring in any S_i has a trace.

Lemma 2. The limit S_ω is saturated up to redundancy.

Lemma 3. The limit S_ω is logically equivalent to the initial set S_0.

Lemma 4. A set S of clauses saturated up to redundancy is unsatisfiable if and only if $\square \in S$.

Proof of Completeness

A trace of a clause C: a set of clauses $\{C_1, \ldots, C_n\} \subseteq S_\omega$ such that

1. $C \succeq C_i$ for all $i = 1, \ldots, n$;
2. $C_1, \ldots, C_n \models C$.

Lemma 1. Every clause C occurring in any S_i has a trace.

Lemma 2. The limit S_ω is saturated up to redundancy.

Lemma 3. The limit S_ω is logically equivalent to the initial set S_0.

Lemma 4. A set S of clauses saturated up to redundancy is unsatisfiable if and only if $\Box \in S$.

Interestingly, only the last lemma uses the rules of $\mathbb{S}\mathrm{up}_{\succ,\sigma}$ and the fact that σ is well-behaving.