Uniform Derivation of Decision Procedures by Superposition

Alessandro Armando, Silvio Ranise,

and Michaël Rusinowitch

$$T \models L_1 \vee L_2 \vee \cdots \vee L_k.$$
$$\Updownarrow$$
$$UnSat(T, \overline{L_1}, \ldots, \overline{L_k})$$

$L_i = (s_i = t_i) \mid (s_i \neq t_i)$ – ground literals

T is decidable $\leftrightharpoons$ $T \models C$ is decidable, C – ground clause

$\Leftrightarrow UnSat_T(S) := UnSat(T \cup S)$ is decidable,

S – set of ground literals

Decision algorithm for T is any algorithm that solves $UnSat_T(S)$.

Example 1

Theory of homomorphisms:[a]

$\mathcal{H}= \{h(f(x,y)) = f(h(x),h(y))\}$

Consider:

$$S := \{h(c) = c',\ h(c') = c,\ f(c,c') = h(h(a)),\ f(c',c) = a,\ h(h(h(a))) \neq a\};$$

$$C := [h(c) = c', h(c') = c, f(c,c') = h(h(a)), f(c',c) = a] \Rightarrow h(h(h(a))) = a;$$

(we abbreviate $L_1, \ldots, L_k \Rightarrow L_{k+1}, \ldots, L_s$ for $\overline{L_1} \vee, \ldots, \overline{L_k} \vee L_{k+1} \vee \cdots \vee L_s$)

Clearly $T \models C$, so S is $\mathcal{H}$-unsatisfiable.

[a] Actually, $\mathcal{H}$ is decidable

Superposition calculus $\mathcal{SP}$

Inference rules:

Superposition $$\frac{\Gamma \Rightarrow \Delta, l[u'] = r \quad \Pi \Rightarrow \Sigma, u = v}{\sigma(\Gamma, \Pi \Rightarrow \Delta, \Sigma, l[v] = r)}$$

Paramodulation $$\frac{\Gamma, l[u'] = r \Rightarrow \Delta \quad \Pi \Rightarrow \Sigma, u = v}{\sigma(l[v] = r, \Gamma, \Pi \Rightarrow \Delta, \Sigma)}$$

Reflection $$\frac{\Gamma, u' = u \Rightarrow \Delta}{\sigma(\Gamma \Rightarrow \Delta)}$$

Factoring $$\frac{\Gamma \Rightarrow \Delta, u = t, u' = t'}{\sigma(\Gamma, t = t' \Rightarrow \Delta, u = t')}$$

$\sigma = MGU(u, u')$, u' is not a variable in *Superposition* and *Paramodulation*.

Superposition calculus $\mathcal{SP}$

Simplification rules:

Subsumption $$\frac{S \cup \{C, C'\}}{S \cup \{C\}} \qquad \exists\ \theta(C) \subseteq C',\ \nexists\ \rho(C') = C$$

Simplification $$\frac{S \cup \{C[\theta(l)], l = r\}}{S \cup \{C[\theta(r)], l = r\}}$$

Deletion $$\frac{S \cup \{\Gamma \Rightarrow \Delta, t = t\}}{S}$$

Simplification rules have higher priority than *inference rules*.

Theorem 1.1 *$\mathcal{SP}$ is refutationally complete:*

$$\boxed{UnSat(T) \Rightarrow \forall\ T \vdash^{fair}_{SP} \{\}.}$$

Theorem 1.1 gives us a semi-desicion procedure for T:

$$
\begin{array}{rl}
UnSat(T \cup S) \Rightarrow & \mathcal{SP} \text{ derive empty clause} \\
\text{not } UnSat(T \cup S) \Rightarrow & \mathcal{SP} \text{ may diverge}
\end{array}
$$

For some theories T $\mathcal{SP}$ may give a decision procedure (Example: T - set of ground clauses).

We give an uniform $\mathcal{SP}$-based decision procedures for:

- Quantifier-free theory of equality
- Theory of Lists
- Theory of Arrays
- Theory of Arrays with Extensionality
- Theory of Lists and Arrays
- Theory of Homomorphisms

The scheme of the decision procedure based on $\mathcal{SP}$

1. "Flatten" the the input set of literals S;
2. Show termination of $\mathcal{SP}$ on $\{\Gamma \cup S\}$ by specifying the clauses that may appear during derivation.

Flattening S

flat equality: $f(t_1, \ldots, t_n) = t_0 \mid t_0 = f(t_1, \ldots, t_n)$

distinction: $t_1 \neq t_2$

were t_i is either a variable or an individual constant;

Flat literals $\leftrightharpoons$ flat equality of flat distinction;

Flat clause $\leftrightharpoons$ disjunction of flat literals

Lemma 1.2 *Let T be a theory, S be a set of ground literals. Then there (eff.) exists S' – a finite set set of flat literals, such that*

$$UnSat_T\{S\} \Leftrightarrow UnSat_T S'$$

Example 2 The set from Example 2: $S = \{h(c) = c',\ h(c') = c,$ $\boxed{f(c, c') = h(h(a))}$, $f(c', c) = a$, $\boxed{h(h(h(a))) \neq a}\}$;

Can be flatten to $S' = \{h(c) = c',\ h(c') = c,$ $\boxed{f(c, c') = c_2}$, $f(c', c) = a$, $h(a) = c_1$, $h(c_1) = c_2$, $h(c_2) = c_3$, $\boxed{c_3 \neq a}\}$;

A Decision Procedure for the Quantifier-Free Theory of Equality

Theory $\mathcal{E}$:

Congruence axioms:		substitutivity (Functional monotonicity):
$x = x$	(R)	$x = y \Rightarrow$
$x = y \Rightarrow y = x$	(S)	$f(\ldots x \ldots) = f(\ldots y \ldots)$
$x = y,\, y = z \Rightarrow x = z$	(T)	

Restriction on ordering $\succ$: $f(t_1, \ldots, t_n) \succ c$ for every constant c.

Lemma 2.1 *Let S be a finite set of flat literals. Then $\mathcal{SP}\{S\}$ is finite.*

Proof. Saturation in $\mathcal{SP}$ can produce only flat literals[a] ■

[a] Superposition can never apply to ground flat literals since simplification has higher priority

A Decision Procedure for the Theory of Lists

$$Ax(\mathcal{L}): \quad car(cons(x, y)) = x \qquad (1)$$
$$cdr(cons(x, y)) = y \qquad (2)$$

$$\mathcal{L} \fallingdotseq \mathcal{E} + Ax(\mathcal{L})$$

Lemma 3.1 *Let S be a finite set of flat literals. Then the clauses occurring in $\mathcal{SP}\{L \cup S\}$ can be:*

- *empty clauses*
- *ground flat literals*
- *$Ax(\mathcal{L})$*

Proof. By case analysis of rules and assumptions. ■

Corollary 3.2 *For every set S of flat literals $\mathcal{SP}\{\mathcal{L} \cup S\}$ is finite. (In fact, $|\mathcal{SP}\{\mathcal{L} \cup S\}|$ is polynomial in $|S|$).*

Theorem 3.3 *$\mathcal{SP}$ is a decision procedure for $\mathcal{L}$.*

A Decision procedure for the Theory of Arrays

$$Ax(\mathcal{A}): \quad select(store(a,i,e),i) = e \qquad (3)$$

$$i \neq j \Rightarrow select(store(a,i,e),j) = select(a,j) \qquad (4)$$

$\mathcal{A} \leftrightharpoons \mathcal{E} + Ax(\mathcal{A})$ Restriction: $t \succ s$ if

1) t contains *select* or *store*, but s is not and closed.

2) t is non-constant, s is constant.

Lemma 4.1 *Let S be a finite set of flat literals. The clauses occuring in the saturations of $S \cup Ax(\mathcal{A})$ by $\mathcal{SP}$ can only be:*

- *the empty clauses;*
- *the axioms in $Ax(\mathcal{A})$;*
- *ground flat literals*
- $t \bowtie t' \vee c_1 = c'_1 \vee \cdots \vee c_n = c'_n$, *where* $t \bowtie t' \in \{c' \neq c''\} \cup \{t = t' | t, t' \in \{c_i, select(c_i, i)\}\}$
- $select(c,x) = select(c',x) \vee c_1 = k_1 \vee \cdots \vee c_n = k_n$, *where* $k_i \in \{x, c, c', c_1, c'_1, \ldots, c_n, c'_n\}$

A Decision procedure for the Theory of Arrays

Lemma 4.1 *Let S be a finite set of flat literals. The clauses occuring in the saturations of $S \cup Ax(\mathcal{A})$ by $\mathcal{SP}$ can only be:*

- *the empty clauses;*
- *the axioms in $Ax(\mathcal{A})$;*
- *ground flat literals*
- $t \bowtie t' \vee c_1 = c'_1 \vee \cdots \vee c_n = c'_n$, *where* $t \bowtie t' \in \{c' \neq c''\} \cup \{t = t' | t, t' \in \{c_i, select(c_i, i)\}\}$
- $select(c, x) = select(c', x) \vee c_1 = k_1 \vee \cdots \vee c_n = k_n$, *where* $k_i \in \{x, c, c', c_1, c'_1, \ldots, c_n, c'_n\}$

Corollary 4.2 *For S being a finite set of flat literals. Then every saturation by $\mathcal{SP}$ of $\{\mathcal{A} \cup S\}$ is finite.*

Theorem 4.3 *$\mathcal{SP}$ is a decision procedure for $\mathcal{A}$.*

A Decision procedure for the Theory of Arrays with Extensionality

$\mathcal{A}^s$ is a many sorted version of $\mathcal{A}$.

$\mathcal{A}^s_e \risingdotseq \mathcal{A}^s +$ $\boxed{\forall i.(select(a,i) = select(b,i)) \Rightarrow a = b \qquad (5)}$

Additional assumptions:

- $f \neq select, store,\ f : s_0, \ldots, s_{n-1} \rightarrow s_n$ then $s_i \neq ARRAY$;
- $\Sigma_{\mathcal{A}^e_s}$ is sensible (there is at least one ground term for each sort).

Given S being a set of ground literals we proceed by:

1. "Lifting of inequalities": $t \neq t' \rightarrow \exists i.select(t,i) \neq select(t',i)$;
   Lemma 5.1 *S is $\mathcal{A}^s_e$-satisfiable iff S' is $\mathcal{A}^s$-satisfiable.*
2. "Skolemization:" $\exists i.select(t,i) \neq select(t',i) \rightarrow$
   $select(t, sk(t,t')) \neq select(t', sk(t,t'))$
   Theorem 5.2 *S' is $\mathcal{A}^s$-satisfiable iff S'' is $\mathcal{A}$-satisfiable.*
3. Apply the decision procedure for $\mathcal{A}$.

Combining Decision Procedures for Lists and Arrays

$Ax(\mathcal{U}) = Ax(\mathcal{L}) + Ax(\mathcal{A})$; $\mathcal{U} = Ax(\mathcal{U}) + \mathcal{E}$

Lemma 6.1 *Let S be a finite set of ground flat literals. The clauses in $\mathcal{SP}\{S \cup Ax(\mathcal{U})\}$ can only be of the types, specifying in Lemma 3.1 and Lemma 4.1.*

Lemma 6.2 *All the saturations of $S \cup Ax(\mathcal{U})$ by $\mathcal{SP}$ are finite.*

Theorem 6.3 *$\mathcal{SP}$ is a decision procedure for $\mathcal{U}$.*

A Decision Procedure for the Theory of Homomorphism

Recall: $Ax(\mathcal{H}) \leftrightharpoons$ $$h(f(x_1, \ldots, x_n)) = f(h(x_1), \ldots, h(x_n)) \qquad (6)$$

$\mathcal{H} = Ax(\mathcal{H}) + \mathcal{E}$

Sketch of the Decision procedure[a]

Given a set Ψ of ground literals we:

1. Flatten Ψ

→2. Complete the set of (ground) equalities in Ψ modulo $Ax(\mathcal{H})$ in order to get a rewrite system $\mathcal{R}$.

3. For each inequality $s \neq t$ in Ψ compute the normal forms $s' = s \downarrow_{\mathcal{R}}$, $t' = t \downarrow_{\mathcal{R}}$ w.r.t. $\mathcal{R}$. If there exists $s \neq t \in \Psi$ s.t. $s' = t'$ then Ψ is $\mathcal{H}$-unsatisfiable; otherwise Ψ is $\mathcal{H}$-satisfiable.

---

[a] actually it's an adaptation of the Knuth-Bendix completion procedure:

Completion

Orientation:

We want $\mathcal{SP}$ to generate only ground literals. For this reason we should orient $Ax(\mathcal{H})$ by the following way:

$$h(f(x_1, \ldots, x_n)) \succ f(h(x_1), \ldots, h(x_n))$$

otherwise, *superposition* between

$$f(h(x_1), \ldots, h(x_n)) = h(f(x_1, \ldots, x_n)) \text{ and } h(c_1) = c$$

will give us $f(c, h(x_2), \ldots, h(x_n)) = h(f(c_1, x_2, \ldots, x_n))$.

Problem: $\mathcal{SP}\{h(c) = c, f(c, c') = c, Ax(\mathcal{H})\}$ generates $f(c, h^n(c')) = c \; \forall n \geq 0$.

$\Rightarrow$ We cannot argue that $\mathcal{SP}\{S \cup \mathcal{H}\}$ is finite.

However, decision procedure for $\mathcal{H}$ can be constructed by using a special rule of inference additional to $\mathcal{SP}$ instead of $Ax(\mathcal{H})$ using so-called completion procedure.

Discussion

? what kind of uniformness in applying $\mathcal{SP}$ as a decision procedure?

? is there a decidable theories to which $\mathcal{SP}$ cannot be applied as a decision procedure? [Yes].

? how to check whether $\mathcal{SP}$ can be applied to a theory as a decision procedure?

? is there a universal decision procedure that can be applied to any decidable theory? [Skeptical]

? what is the time complexity of these decision procedures w.r.t. the best time complexities?

? what is the relationship between congruence closure and $\mathcal{SP}$ (applicability, complexity)?