max planck institut informatik

Automated Reasoning Building Blocks

Part 2

Christoph Weidenbach

Max Planck Institute for Informatics

September 21, 2015

Outline

First-Order Theory Reasoning

Fixed Theory Reasoning

Combination of Theories

First-Order Superposition

Superposition no Equality

Superposition Left

$(N \uplus \{C \vee L, D \vee K\}) \Rightarrow_{\mathsf{SUP}} (N \cup \{C \vee L, D \vee K\} \cup \{(C \vee D)\sigma\})$

where (i) L is positive, strictly maximal in $C \vee L$

(ii) no literal in $C \vee L$ is selected

(iii) K is negative, maximal and no literal selected in $D \vee K$, or K is selected in $D \vee K$

(iv) $L\sigma = \mathsf{comp}(K)\sigma$ for mgu σ

Factoring

$(N \uplus \{C \vee L \vee K\}) \Rightarrow_{\mathsf{SUP}} (N \cup \{C \vee L \vee K\} \cup \{(C \vee L)\sigma\})$

where (i) L is positive, maximal in $C \vee L \vee K$

(ii) no literal is selected in $C \vee L \vee K$

(iii) $L\sigma = K\sigma$ for mgu σ

Superposition Results

Theorem (Completeness, Models, Redundancy)

If all superposition inferences in N up to redundancy are performed and $\bot \notin N$ then N is satisfiable and $N_{\mathcal{I}} \models N$.

It is sufficient to consider inferences between a minimal false clause $\neg P \vee C$, $N_{\mathcal{I}} \not\models \neg P \vee C$ and its productive counterpart $P \vee D$.

The result $C \vee D$ of the superposition inference is not redundant.

First-Order Specialities

- $N_{\mathcal{I}}$ is constructed over up to infinitely many ground instances
- $N_{\mathcal{I}} \not\models \neg P \vee C$ undecidable, in general
- unsatsfiability is semi-decidable, so saturation needs not terminate

Concrete First-Order Redundancy

Subsumption
$(N \uplus \{C, D\}) \Rightarrow_{\mathsf{SUP}} (N \cup \{C\})$
provided $C\sigma \subset D$

Condensation
$(N \uplus \{C \vee L \vee K\}) \Rightarrow_{\mathsf{SUP}} (N \cup \{C \vee L\})$
provided $L\sigma = K$, $(C \vee L)\sigma$ subsumes $C \vee L \vee K$

Tautology Deletion
$(N \uplus \{C \vee L \vee \mathrm{comp}(L)\}) \Rightarrow_{\mathsf{SUP}} (N)$

Subsumption Resolution
$(N \uplus \{C_1 \vee K, C_2 \vee L\}) \Rightarrow_{\mathsf{SUP}} (N \cup \{C_1 \vee L, C_2\})$
where $C_1\sigma \subseteq C_2$, $K\sigma = \mathrm{comp}(L)$

Linear Monadic Shallow Horn Theories

LMSH Theories [W1999]

- Linear: positive literals $C \vee S(f(x, y))$ $\quad C \vee S(f(x, x))$
- Monadic: all literals $C \vee (\neg)S(f(x, y))$ $\quad C \vee (\neg)R(f(x, y), z)$
- Shallow: positive literals $C \vee S(f(x, y))$ $\quad C \vee S(f(x, g(y)))$
- Horn: positive literals $C \vee S(f(x, y))$ $\quad C \vee S(g(y)) \vee T(y)$

Example (LMSH Theories)

- $\neg S(x) \vee \neg T(x) \vee Q(x)$
- $\neg S(x) \vee \neg T(x) \vee Q(f(x, y))$
- $C \vee (\neg)Q(f(a, g(b)))$
- $C \vee \neg Q(f(x, (f(x, g(y)))))$

LMSH Properties

Approximation [TeuckeW2015]

- Linear: fresh variables
  $\neg P(x) \vee S(f(x, x)) \;\Rightarrow\; \neg P(x) \vee \neg P(y) \vee S(f(x, y))$
- Monadic: functions
  $C \vee (\neg)R(f(x, y), z) \;\Rightarrow\; C \vee (\neg)T(f_R(f(x, y), z))$
- Shallow: fresh predicates
  $C \vee S(f(x, g(y))) \;\Rightarrow\; C' \vee \neg P(z) \vee S(f(x, z)); C'' \vee P(g(y))$
- Horn: pick atom
  $C \vee S(f(x, y)) \vee T(g(y)) \;\Rightarrow\; C \vee S(f(x, y))$

Theorem (LMSH [W1999, ReussSeidel2010])

Satisfiability of LMSH (and extensions) is decidable.

Superposition Selection Strategy

$S(a)^*$ $\quad$ (*means maximal)
$\neg S(x)^+ \vee S(g(x))$ $\quad$ ($^+$means selected)
$\Rightarrow_{\mathsf{SUP}} S(g(a))^* \Rightarrow_{\mathsf{SUP}} S(g(g(a)))^* \Rightarrow_{\mathsf{SUP}} \ldots$
no $S(g^i(a))$ redundant, because $\neg S(g^{i-1}(a)) \vee S(g^i(a)) \succ S(g^i(a))$

$S(a)^*$
$\neg S(x) \vee S(g(x))^*$
saturated

Superposition Selection Strategy

$\neg R(g(x)) \vee S(f(x,y))^*$
$\neg S(f(g(x),y))^* \vee \neg S(y)$
$\Rightarrow_{\mathsf{SUP}} \neg R(g(g(x))) \vee \neg S(y)$

$\neg R(g(x))^+ \vee S(f(x,y))$
$\neg S(f(g(x),y))^+ \vee \neg S(y)$
$\neg T(x) \vee \neg T(y) \vee S(f(x,y))^*$
$\Rightarrow_{\mathsf{SUP}} \neg T(g(x)) \vee \neg T(y) \vee \neg S(y)$

LMSH Termination

Selection Strategy

Select a negative literal $\neg S_i(t_i)$ in $\neg S_1(t_1) \vee \ldots \neg S_n(t_n) \vee S(s)$ if

- t_i is not a variable
- all t_j variables, t_i does not occur in s

Termination

- all derived clauses bound in term depth
- no variable chains
- only finitely many different clauses, with respect to subsumption and condensation

LMSH Saturation Clauses

Positive Maximal

- $\neg S_{i_1}(x_{j_1}) \vee \ldots \vee \neg S_{i_n}(x_{j_n}) \vee S(f(x_1,\ldots,x_m))^*$
- $\neg S_1(x) \vee \ldots \vee \neg S_n(x) \vee S(x)^*$

Negative Selected/Maximal

- $\neg S_1(x) \vee \ldots \neg S_i(x)^* \ldots \vee \neg S_n(x) \vee S(x)$
- $\neg S_1(x) \vee \ldots \neg S_i(x)^+ \ldots \vee \neg S_n(x) \vee S(t) \quad x \notin \mathsf{vars}(t)$
- $\neg S(f(t_1,\ldots,t_n))^+ \vee C$

Bernays-Schönfinkel (BS) Decidability

BS Class

First-Order clauses where all functions are constants.

Example (BS Clauses)

Father(John, Peter)
$\neg$Ancestor$(x,y) \vee \neg$Ancestor$(y,z) \vee$ Ancestor(x,z)
$\neg$Relative$(x,y) \vee$ Relative(y,x)

Decision Procedures

Model Evolution [BaumgartnerEtEl2006]
SAT Instantiation [GanzingerKorovin2003]
Non-Redundant Clause Learning [AlagiW2015]

BS Horn Superposition

Example (Transitivity)

- $\neg\text{Ancestor}(x, y) \vee \neg\text{Ancestor}(y, z) \vee \text{Ancestor}(x, z)$
- assume a reasonable ontology with 10^6 individuals
- then transitivity has $(10^6)^3$ different instances
- to be explored by any explicit model building calculus

Transitivity Horn Superposition [BG94Trans]

PChaining

$(N \cup \{Q(l, s),\ Q(t, r)\}) \Rightarrow_{\text{TSUP}} (N \cup \{Q(l, r)\sigma\})$

provided $s\sigma = t\sigma$, $l\sigma \not\succeq s\sigma$ and $r\sigma \not\succeq t\sigma$

NChaining1

$(N \cup \{Q(l, s), C \vee \neg Q(t, r)\}) \Rightarrow_{\text{TSUP}} (N \cup \{(C \vee \neg Q(t, l))\sigma\})$

provided $s\sigma = r\sigma$, $s\sigma \not\succeq l\sigma$ and $r\sigma \not\succeq t\sigma$

Transitivity Superposition on YAGO

The YAGO theory has about 10^6 individuals, 500.000 non-unit clauses, 9.500.000 ground facts.

SPASSYAGO Saturation [SudaEtAl2010]

Slices	Input Clauses	Time	Output Clauses
S_0	136808	12.5	+1768
S_1	132080	9.7	+16060
S_2	96454	9.9	+1768
S_3	114527	10.6	+4769
S_4	4891235	37:11.1	+24123
Full	9918933	1:03:24.0	+24123

There is no other system capable of showing satisfiablity of YAGO.

Superposition Decision Procedures

Classes

- HSML [W1999, ReussSeidel2010]
- Monadic Class with Equality [BGW93]
- Guarded Fragment with Equality [GanzingerNivelle99]
- Shallow Equational [JacquemardEtAl98]
- Bounded Domains [HillenbrandW13]
- Tree Automata with Equality Constraints [JacquemardRV06]
- Saturated Clause Sets [HorbachW09]
- Data Structures [ArmandoEtAl09]
- Modal Logics [SchmidtHustadt13]

Summary

Take Aways

- Superposition is an Effective and Efficient Decision Procedure for Many Classes
- For an Efficient Decision Procedure Model Building but also Saturation can be Key

References HSML and Sup

- Andreas Reuß and Helmut Seidl.
  Bottom-up tree automata with term constraints.
  In LPAR 2010, LNCS 6397, pages 581–593. 2010.
- Andreas Teucke and Christoph Weidenbach.
  First-order logic theorem proving and model building via approximation and instantiation.
  In FroCoS'15, LNAI 9322. 2015.
- Christoph Weidenbach.
  Towards an automatic analysis of security protocols in first-order logic.
  In CADE-16, LNAI 1632, pages 314–328. 1999.
- Leo Bachmair and Harald Ganzinger.
  Rewrite techniques for transitive relations.
  In LICS 1994, pages 384–393. 1994.

References BS

- Gabor Alagi and Christoph Weidenbach.
  NRCL - a model building approach to the Bernays-Schönfinkel fragment.
  In FroCoS'15, LNAI 9322. 2015.
- Peter Baumgartner, Alexander Fuchs, and Cesare Tinelli.
  Lemma learning in the model evolution calculus.
  In LPAR'2006, LNCS 4246, pages 572–586. 2006.
- Harald Ganzinger and Konstantin Korovin.
  New directions in instantiation-based theorem proving.
  In LICS'2003, pages 55–64. 2003.
- Martin Suda, Christoph Weidenbach, and Patrick Wischnewski.
  On the saturation of YAGO.
  In IJCAR 2010, LNAI 6173, pages 441–456, 2010.

References Other Classes

- Leo Bachmair, Harald Ganzinger, and Uwe Waldmann.
  Superposition with simplification as a decision procedure for the monadic class with equality.
  In Computational Logic and Proof Theory, LNCS 713, pages 83–96. 1993.
- Harald Ganzinger and Hans de Nivelle.
  A superposition decision procedure for the guarded fragment with equality.
  In LICS, pages 295–304, 1999.
- Florent Jacquemard, Christoph Meyer, and Christoph Weidenbach.
  Unification in extensions of shallow equational theories.
  In RTA-98, LNCS 1379, pages 76–90. 1998.

References Other Classes

- Thomas Hillenbrand and Christoph Weidenbach.
  Superposition for bounded domains.
  In McCune Festschrift, LNCS 7788, pages 68–100. 2013.
- Florent Jacquemard, Michaël Rusinowitch, and Laurent Vigneron.
  Tree automata with equality constraints modulo equational theories.
  In IJCAR 2006, LNCS 4130, pages 557–571. 2006.
- Matthias Horbach and Christoph Weidenbach.
  Decidability results for saturation-based model building.
  In CADE 2009, LNCS 5663, pages 404–420. 2009.

References Other Classes

- Alessandro Armando, Maria Paola Bonacina, Silvio Ranise, and Stephan Schulz.
  New results on rewrite-based satisfiability procedures.
  ACM Transactions on Computational Logic, 10(1):4:1–4:51, 2009.
- Renate A. Schmidt and Ullrich Hustadt.
  First-order resolution methods for modal logics.
  In LNCS 7797, pages 345–391. 2013.

EUF: Equality with Uninterpreted Functions

An equational formula

$\forall \vec{x}(t_1 \approx s_1 \wedge \ldots \wedge t_n \approx s_n \rightarrow u_1 \approx v_1 \vee \ldots \vee u_m \approx v_m)$

is valid iff its Skolemized negation

$(t_1 \approx s_1 \wedge \ldots \wedge t_n \approx s_n \wedge u_1 \not\approx v_1 \vee \ldots \vee u_m \not\approx v_m)\{\vec{x} \mapsto \vec{c}\}$

is unsatisfiable.

Idea: turn the equations $t_i \approx s_i$ into a convergent rewrite system R and then test the $u_i \downarrow_R \neq v_i \downarrow_R$.

Trick: flatten all terms by introducing new constants.
Effect: sharing: a unique name for every subterm

Flattening

The equation $g(a, h(h(b))) \approx h(a)$ is replaced by

$h(b) \approx c_3 \wedge h(c_3) \approx c_4 \wedge g(a, c_4) \approx c_5 \wedge h(a) \approx c_6 \wedge c_5 \approx c_6$

After the transformation only two types of equations are left:
$f(\vec{c}) \approx c_j$
$c_i \approx c_j$

The congruence closure algorithm starts with a set of flattened equations and an empty rewrite system: $(E, \emptyset)$

Congruence Closure

Rules

Simplify
$(E \cup \{c \dot{\approx} d\}; R \cup \{c \rightarrow e\}) \Rightarrow_{CC} (E \cup \{e \dot{\approx} d\}; R \cup \{c \rightarrow e\})$

Delete
$(E \cup \{c \approx c\}; R) \Rightarrow_{CC} (E; R)$

Orient
$(E \cup \{c \dot{\approx} d\}; R) \Rightarrow_{CC} (E; R \cup \{c \rightarrow d\})$
if $c \succ d$

Deduce
$(E; R \cup \{t \rightarrow c,\ t \rightarrow d\}) \Rightarrow_{CC} (E \cup \{c \approx d\}; R \cup \{t \rightarrow c\})$

Collapse
$(E; R \cup \{t[c]_p \rightarrow d,\ c \rightarrow e\}) \Rightarrow_{CC} (E; R \cup \{t[e]_p \rightarrow d,\ c \rightarrow e\})$
$p \neq \epsilon$

Linear Arithmetic

Conjunctions of atoms over the signature $\{+, -, \leq, <, \approx, \not\approx, >, \geq\}$ and interpreted over $\mathbb{Q}$.

For example:
$x + y \geq 16 \wedge 4x + 7y \leq 28 \wedge 2x - 7y \leq 20 \wedge 2x - 3y \geq -9$

Fourier-Motzkin

$$\exists x \left(\bigwedge_i x < s_i(\vec{y}) \ \wedge \ \bigwedge_j x \leq t_j(\vec{y}) \ \wedge \ \bigwedge_k x > u_k(\vec{y}) \ \wedge \ \bigwedge_l x \geq v_l(\vec{y}) \ \wedge \ \bigwedge_m 0 \sim_m w_m(\vec{y}) \right)$$

$$\Rightarrow_{\mathsf{FM}}$$

$$\bigwedge_i \bigwedge_k s_i(\vec{y}) > u_k(\vec{y}) \ \wedge \ \bigwedge_j \bigwedge_k t_j(\vec{y}) > u_k(\vec{y}) \ \wedge \ \bigwedge_i \bigwedge_l s_i(\vec{y}) > v_l(\vec{y}) \ \wedge \ \bigwedge_j \bigwedge_l t_j(\vec{y}) \geq v_l(\vec{y}) \ \wedge \ \bigwedge_m 0 \sim_m w_m(\vec{y})$$

where $\sim_j \in \{<, >, \leq, \geq\}$

CDCL(T)

Extra Rules

T-Success
$(M; N; U; k; \top) \ \Rightarrow_{\mathsf{CDCL(T)}} \ (M; \emptyset; N \cup U; k; \top)$
provided all literals in N defined, M is T-satisfiable and $M \models_T N$

T-Conflict
$(M; N; U; k; \top) \ \Rightarrow_{\mathsf{CDCL(T)}} \ (M; N; U; l; L_1 \vee \ldots \vee L_n)$
provided $\mathsf{comp}(L_i) \in M$, $\mathsf{comp}(L_1), \ldots, \mathsf{comp}(L_n) \models_T \bot$ and l is the maximal level of an L_j

T-Propagate
$(M; N; U; k; \top) \ \Rightarrow_{\mathsf{CDCL(T)}} \ (ML^{C \vee L}; N; U; k; \top)$
provided L is undefined, $L_i \in M$, M is T-satisfiable , $L_1, \ldots, L_n \models_T L$ and $C = \mathsf{comp}(L_1) \vee \ldots \vee \mathsf{comp}(L_n)$

Motivation

Arithmetic

- Arithmetic is not (completely) first-order axiomatizable.
- Even for an axiomatizable fragment there is no decision procedure.
- Build in approaches analogous to equality are not satisfactory.

First-Order Logic Modulo Arithmetic

- Start with standard LA= $(\mathbb{Q}, +, -, \{<, \leq, >, \geq, =, \neq\})$, vars $\mathcal{X}_{\mathbb{Q}}$
- Extend with additional FOL predicate and function symbols Σ
- Add FOL variables of sort $\top$: $\mathcal{X} = \mathcal{X}_{\mathbb{Q}} \cup \mathcal{X}_{\top}$

Example: Program Analysis

while Program Analysis

1	$n = 0;$	$\neg P_1(n,x,y) \vee P_2(0,x,y)$
2	while $(x > 0)$ {	$x > 0 \parallel \neg P_2(n,x,y) \vee P_3(n,x,y)$
		$x \leq 0 \parallel \neg P_2(n,x,y) \vee P_6(n,x,y)$
3	$\quad n = n + y;$	$\neg P_3(n,x,y) \vee P_4(n+y,x,y)$
4	$\quad x = x - 1;$	$\neg P_4(n,x,y) \vee P_5(n,x-1,y)$
5	}	$\neg P_5(n,x,y) \vee P_2(n,x,y)$
6	return n;	

Example: Time in Ontologies

Marriage: No Time

- Married(John, Mary)
- $\neg$Married$(x,y) \vee \neg$Married$(x,z) \vee y \approx z$
- Married(John, Jill)

Marriage: Time

- $5 \leq u \leq 15 \parallel$ Married$(u$, John, Mary$)$
- $\neg$Married$(u,x,y) \vee \neg$Married$(u,x,z) \vee y \approx z$
- $16 \leq u \parallel$ Married$(u$, John, Jill$)$

About Complexity

The Naturals

- $N(0)$
- $\neg N(u) \vee N(u+1)$
- $0 < u < 1 \parallel \neg N(u)$
- $u > 0 \parallel \neg N(u+1) \vee N(u)$
- $u < 0 \parallel \neg N(u)$

Theorem (FOL(Σ,LA) undecidability)

Unsatisfiability of a FOL(Σ,LA) clause set is undecidable.

About Compactness

The Naturals Extended

- $\neg P(a)$
- $N(a)$
- $\neg N(u) \vee \neg P(u) \vee P(u+1)$
- $P(0)$

Theorem (FOL(Σ,LA) not compact)

- *There are FOL(Σ,LA) clause sets where a finite set of ground instances is satisfiable, an infinite set is not.*
- *There cannot be a complete calculus for FOL(Σ,LA) clause sets, in general.*

Expressivity

Theorem ($\mathbb{Q}$ and $\mathbb{R}$ cannot be distinguished [Skolem20])

Every FOL(LA) formula has a model over $\mathbb{Q}$ iff it has a model over $\mathbb{R}$.

Example (FOL(Σ,LA))

$x = y \lor g(x) \geq g(y) + 1 \lor g(x) + 1 \leq g(y)$

Nelson Oppen Theory Combination

Restrictions [NelsonOpppen79, GhilardiEtAl08, TinelliRingeissen03]

- decision procedures for T_1, T_2
- complete for variable (constant) identities
- disjoint signatures except for variables (constants)
- all models of T_1, T_2 are infinite (Horn, convex)

Example

T_1 is EUF and T_2 is linear integer arithmetic over $\mathbb{Q}$.
Is $x - y \approx 0 \land g(x) \not\approx g(y)$ satisfiable?

Nelson-Oppen States

Idea: The theories T_i with purified problems N_i communicate via implied equalities E_i.

$(N_1; \emptyset; N_2; \emptyset; \bot)$	start state for set of purified atoms
$(N_1; E_1; N_2; E_2; \text{fail})$	final state, unsatisfiable
$(N_1; E_1; N_2; E_2; \bot)$	intermediate state
$(N_1; \emptyset; N_2; \emptyset; \top)$	final state, where $N_1 \cup N_2$ is satisfiable

Nelson-Oppen Rules

Solve
$(N_1; E_1; N_2; E_2; \bot) \Rightarrow_{\text{NO}} (N_1'; E_1'; N_2'; E_2'; \bot)$
if $N_1' = N_1 \cup E_1 \cup E_2$ and $N_2' = N_2 \cup E_1 \cup E_2$ are both T_i-satisfiable, respectively, E_1' are all new variable equations derivable from N_1', E_2' are all new variable equations derivable from N_2' and $E_1' \cup E_2' \neq \emptyset$

Success
$(N_1; E_1; N_2; E_2; \bot) \Rightarrow_{\text{NO}} (N_1'; \emptyset; N_2'; \emptyset; \top)$
if $N_1' = N_1 \cup E_1 \cup E_2$ and $N_2' = N_2 \cup E_1 \cup E_2$ are both T_i-satisfiable, respectively, E_1' are all new variable equations derivable from N_1', E_2' are all new variable equations derivable from N_2' and $E_1' \cup E_2' = \emptyset$

Fail
$(N_1; E_1; N_2; E_2; \bot) \Rightarrow_{\text{NO}} (N_1; E_1; N_2; E_2; \text{fail})$
if $N_1' = N_1 \cup E_1 \cup E_2$ or $N_2' = N_2 \cup E_1 \cup E_2$ is T_i-unsatisfiable, respectively

First Example

The already purified EUF, LA unsatisfiable combination problem

$x - y \approx 0 \wedge g(x) \not\approx g(y)$

is refuted as follows

$$
\begin{aligned}
&(\{x - y \approx 0\}, \emptyset, \{g(x) \not\approx g(y)\}, \emptyset, \bot)\\
&\Rightarrow^{\text{Solve}}_{\text{NO}} (\{x - y \approx 0\}, \{x \approx y\}, \{g(x) \not\approx g(y)\}, \emptyset, \bot)\\
&\Rightarrow^{\text{Fail}}_{\text{NO}} (\{x - y \approx 0\}, \{x \approx y\}, \{g(x) \not\approx g(y)\}, \emptyset, \text{fail})
\end{aligned}
$$

Second Example

The LA, EUF formula $f(x_1, 0) \geq x_3 \wedge f(x_1, 0) \leq x_3$

becomes after purification

$x_4 \geq x_3 \wedge f(x_1, x_5) \approx x_4 \wedge x_5 \approx 0 \wedge x_6 \leq x_3 \wedge f(x_1, x_5) \approx x_6$

and the respective Neslon-Oppen derivation is

$$
\begin{aligned}
&(\{x_4 \geq x_3, x_5 \approx 0, x_6 \leq x_3\}, \emptyset, \{f(x_1, x_5) \approx x_4, f(x_1, x_5) \approx x_6\}, \emptyset, \bot)\\
&\Rightarrow^{\text{Solve}}_{\text{NO}} (\{x_4 \geq x_3, x_5 \approx 0, x_6 \leq x_3\}, \emptyset,\\
&\qquad \{f(x_1, x_5) \approx x_4, f(x_1, x_5) \approx x_6\}, \{x_4 \approx x_6\}, \bot)\\
&\Rightarrow^{\text{Solve}}_{\text{NO}} (\{x_4 \approx x_6, x_4 \geq x_3, x_5 \approx 0, x_6 \leq x_3\}, \{x_4 \approx x_3, x_6 \approx x_3\},\\
&\qquad \{f(x_1, x_5) \approx x_4, f(x_1, x_5) \approx x_6, x_4 \approx x_6\}, \emptyset, \bot)\\
&\Rightarrow^{\text{Success}}_{\text{NO}} (\{x_4 \approx x_6, x_4 \geq x_3, x_5 \approx 0, x_6 \leq x_3, x_4 \approx x_3, x_6 \approx x_3\}, \emptyset,\\
&\qquad \{f(x_1, x_5) \approx x_4, f(x_1, x_5) \approx x_6, x_4 \approx x_6, x_4 \approx x_3, x_6 \approx x_3\}, \emptyset, \top)
\end{aligned}
$$

Superposition Modulo LA

The FOL Rules without Equality

Superposition Left

$(N \cup \{\Lambda \parallel C \vee L, \Gamma \parallel D \vee K\}) \Rightarrow_{\text{SUP(LA)}} (N \cup \{(\Lambda, \Gamma \parallel C \vee D)\sigma\})$

where (i) L is positive, strictly maximal in $C \vee L$
(ii) no literal in $C \vee L$ is selected
(iii) K is negative, maximal and no literal selected in $D \vee K$, or
K is selected in $D \vee K$
(iv) $L\sigma = \text{comp}(K)\sigma$ for mgu σ

Factoring

$(N \cup \{\Lambda \parallel C \vee L \vee K\}) \Rightarrow_{\text{SUP(LA)}} (N \cup \{(\Lambda \parallel C \vee L)\sigma\})$

where (i) L is positive, maximal in $C \vee L \vee K$
(ii) no literal is selected in $C \vee L \vee K$
(iii) $L\sigma = K\sigma$ for mgu σ

Superposition Modulo LA

The LA Rule

Constraint Refutation

$(N \cup \{\Lambda_1 \parallel \bot, \ldots, \Lambda_n \parallel \bot\}) \Rightarrow_{\text{SUP(LA)}} (N \cup \{\bot\})$

where $\neg\Lambda_1 \wedge \ldots \wedge \neg\Lambda_n$ is false

$N = \{u > 5 \parallel P(u),\ u > 6 \parallel \neg P(u)\}$
$\Rightarrow_{\text{SUP(LA)}} u > 5, u > 6 \parallel \bot$
$\Rightarrow_{\text{SUP(LA)}} \bot$ because $\forall u.(u \leq 5 \vee u \leq 6)$ is false

$N = \{u = 5 \parallel P(u),\ u = 6 \parallel \neg P(u)\}$
$\Rightarrow_{\text{SUP(LA)}} u = 5, u = 6 \parallel \bot$
no inference because $\forall u.(u \neq 5 \vee u \neq 6)$ is true

SUP(LA) Redundancy

Subsumption

Subsumption
$(N \uplus \{\Lambda \parallel C, \Gamma \parallel D\}) \Rightarrow_{\mathrm{SUP(LA)}} (N \cup \{\Lambda \parallel C\})$
provided $C\sigma \subset D$ and $\forall \vec{v}.(\Gamma \rightarrow \Lambda\sigma)$ is true

$u > 5 \parallel P(u)$ subsumes $v > 6 \parallel P(v) \vee Q(v)$
$\sigma = \{u \mapsto v\}$
$\forall v.(v > 6 \rightarrow v > 5)$ is true

$u > 6 \parallel P(u)$ does not subsume $v > 5 \parallel P(v) \vee Q(v)$
$\sigma = \{u \mapsto v\}$
$\forall v.(v > 5 \rightarrow v > 6)$ is false

SUP(LA) Decidability

Theorem (Ground (SMT) Case [KruglovWeidenbach12])

SUP(LA) is a decision procedure for sets of ground clauses.

$N = \{a \neq 5 \parallel P(a), a \neq 5 \parallel \neg P(a), a = 5 \parallel Q(a), a = 5 \parallel \neg Q(a)\}$
$\Rightarrow_{\mathrm{SUP(LA)}} a \neq 5 \parallel \bot$
$\Rightarrow_{\mathrm{SUP(LA)}} a = 5 \parallel \bot$
$\Rightarrow_{\mathrm{SUP(LA)}} \bot$ because $\exists a.(a = 5 \wedge a \neq 5)$ is false

Constraint Refutation Condition in Detail
$\exists \vec{a}.(\forall \vec{v_1}.(\neg\Lambda_1) \wedge \ldots \wedge \forall \vec{v_n}.(\neg\Lambda_n))$ is false

SUP(LA) Decidability

Theorem (Timed Automata[FW12, FietzkeEtAl12])

SUP(LA) is a decision procedure for timed automata reachability.

$u' = 0, v' = 0, u' \leq 4 \parallel \neg R(l_0, u, v) \vee R(l_1, u', v')$
$t \geq 0, u'' = u' + t, v'' = v' + t, u'' \leq 4 \parallel \neg R(l_1, u', v') \vee R(l_1, u'', v'')$
$u' = 0, v' = v, u \geq 2, u' \leq 4 \parallel \neg R(l_1, u, v) \vee R(l_1, u', v')$
$u' = u, v' = v, v \leq 10 \parallel \neg R(l_1, u, v) \vee R(l_2, u', v')$

Summary

Take Aways: Nelson-Oppen

- standard combination procedure in SMT solvers
- Nelson-Oppen modulo CDCL(T)

Take Aways: Superposition

- Hierarchic Superposition is a very powerful calculus for FOL(XXX) theories.
- Implementation techniques are at the start
- Model Building is at the start

References LA and SUP(LA)

- Leo Bachmair, Harald Ganzinger, and Uwe Waldmann.
  Refutational theorem proving for hierarchic first-order theories.
  AAECC, 5(3/4):193–212, 1994.
- Evgeny Kruglov and Christoph Weidenbach.
  Superposition decides the first-order logic fragment over ground theories.
  MCS, 6(4):427–456, 2012.
- Thoralf Skolem.
  Logisch-kombinatorische Untersuchungen über die Erfüllbarkeit oder Beweisbarkeit mathematischer Sätze nebst einem Theoreme über dichte Mengen.
  Skrifter utgit av Videnskappsellkapet i Kristiania, 4:4–36, 1920.
  Translation: Jean van Heijenoort, Harvard Press, 1976.

References SUP(LA)

- Arnaud Fietzke and Christoph Weidenbach.
  Superposition as a decision procedure for timed automata.
  MCS, 6(4):409–425, 2012.
- Arnaud Fietzke, Evgeny Kruglov, and Christoph Weidenbach.
  Automatic generation of Invariants for Circular Derivations in SUP(LA).
  In LPAR 2012, LNCS 7180, pages 197–211. 2012.

References Nelson-Oppen

- G. Nelson and D.C. Oppen.
  Simplification by cooperating decision procedures.
  ACM Transactions on Programming Languages and Systems, 1(2):245–257, 1979.
- Silvio Ghilardi, Enrica Nicolini, and Daniele Zucchelli.
  A comprehensive combination framework.
  ACM Trans. Comput. Log., 9(2), 2008.
- Cesare Tinelli and Christophe Ringeissen.
  Unions of non-disjoint theories and combinations of satisfiability procedures.
  Theoretical Computer Science, 290(1):291–353, 2003

Summary Summary

Automated Reasoning Building Blocks

- Consider Theories: dedicated solvers, combinations
- Be Small: aggressively apply redundancy
- Don't Guess: instead explore candidate model
- Always Learn: CDCL(T), Superposition
- Eliminate Redundancy: CDCL(T), Superposition
- Consider Ordering: proof ordering, term ordering, model ordering
- Learn Fresh: model-based inferences are non-redundant
- Consider Models: model building CDCL(T), decidable theories
- Flexible Models: adapt candidate model to problem

Thanks for Your Attention!