Semantically-guided goal-sensitive reasoning: theorem proving and decision procedures[1]

Maria Paola Bonacina

Dipartimento di Informatica, Università degli Studi di Verona,
Verona, Italy, EU

Dagstuhl Seminar 21371 on Integrated Deduction, 13 September 2021

(Subsuming "SGGS decision procedures for fragments of first-order logic," Dagstuhl Seminar 21361 on Extending the Synergies between SAT and Description Logics, 7 September 2021, and "Conflict-driven first-order decision procedures," Theoretical Foundations of SAT/SMT Workshop, Satisfiability: Theory, Practice and Beyond Program, Simons Institute, 24 March 2021)

[1] Based on joint work with Sarah Winkler and joint work with David A. Plaisted

SGGS: Semantically Guided Goal Sensitive reasoning

SGGS decision procedures

The Koala prover and experimental results

Discussion

Setting the stage

- Decidability of satisfiability + expressivity: decidable FOL fragments
- Refutationally complete inference system for ATP
- Show that it is guaranteed to halt on all inputs in fragment $\mathcal{F}$
- Any fair strategy with that inference system is a decision procedure for satisfiability in $\mathcal{F}$

Apply this approach to SGGS

Why SGGS?

- Model-based: search for a model by building candidates represented by a trail Γ of clauses
  SGGS-derivation: a series of trails
- Conflict-driven: apply resolution mostly to explain conflicts
- SGGS as a first-order analogue of CDCL
- Semantically-guided: fixed initial Herbrand interpretation $\mathcal{I}$
  In this talk: $\mathcal{I}$ is $\mathcal{I}^+$ (all positive) or $\mathcal{I}^-$ (all negative)
- Model complete in the limit: for a satisfiable input the limit of any fair derivation represents a model
- SGGS decision procedures are model-constructing

Model representation in SGGS

- S: set of clauses
- $\mathcal{I} \not\models S$: search for a model
- Γ: trail of clauses $A \rhd C[L]$ where literal L is selected
  A: a kind of Herbrand constraints ($x \not\equiv y$, $top(x) \neq f$)
- Partial model $\mathcal{I}^p(\Gamma)$: each clause adds the ground instances $L\sigma$ s.t. $C\sigma$ not satisfied and $\neg L\sigma$ not already in
- Model $\mathcal{I}[\Gamma]$: complete $\mathcal{I}^p(\Gamma)$ by consulting $\mathcal{I}$
- Get a Γ with either $\perp$ or $\mathcal{I}[\Gamma] \models S$

Example I (part I)

- S contains $\{\ P(a),\ \neg P(x) \vee Q(f(y)),\ \neg P(x) \vee \neg Q(z)\ \}$
- $\mathcal{I}$ is $\mathcal{I}^-$ (all-negative)
- Γ_0 is empty: $\mathcal{I}[\Gamma_0] = \mathcal{I} \not\models P(a)$
- $\Gamma_1 = [P(a)]$ by SGGS-extension
- $\mathcal{I}[\Gamma_1] \not\models \neg P(x) \vee Q(f(y))$
- $\Gamma_2 = [P(a)],\ \neg P(a) \vee [Q(f(y))]$
  by SGGS-extension with mgu $\alpha = \{x \leftarrow a\}$
  where $\neg P(a)$ is assigned to $[P(a)]$

SGGS invariants I

- Literal L is uniformly false in interpretation $\mathcal{J}$ if $\mathcal{J} \models \neg L$
- Every literal in Γ must be $\mathcal{I}$-true ($\mathcal{I} \models L$) or $\mathcal{I}$-false ($\mathcal{I} \models \neg L$)
- If a clause in Γ has $\mathcal{I}$-false literals, one must be selected
- An $\mathcal{I}$-true literal is selected only if all literals in the clause are $\mathcal{I}$-true: $\mathcal{I}$-all-true clause
- Disjoint prefix $dp(\Gamma)$: longest prefix where every selected literal contributes to $\mathcal{I}[\Gamma]$ all its ground instances (no intersection of selected literals)

SGGS invariants II

- $\mathcal{I}$-true literal L in C_i made uniformly false in $\mathcal{I}[\Gamma]$ by the selection of $\mathcal{I}$-false literal M in C_j $(j < i)$: L assigned to C_j
- Non-selected $\mathcal{I}$-true literals must be assigned
- Selected $\mathcal{I}$-true literals must be assigned if possible
- If assigned, a selected $\mathcal{I}$-true literal is assigned rightmost

Example I (part II)

- S contains $\{ P(a),\ \neg P(x) \vee Q(f(y)),\ \neg P(x) \vee \neg Q(z) \}$
- $\Gamma_2 = [P(a)],\ \neg P(a) \vee [Q(f(y))]$
- $\mathcal{I}[\Gamma_2] \not\models \neg P(x) \vee \neg Q(z)$
- $\Gamma_3 = [P(a)],\ \neg P(a) \vee [Q(f(y))],\ \neg P(a) \vee [\neg Q(f(y))]$
  by SGGS-extension with mgu $\alpha = \{x \leftarrow a, z \leftarrow f(y)\}$
  where $\neg P(a)$ is assigned to $[P(a)]$ and $\neg Q(f(y))$ to $[Q(f(y))]$
- Conflict: $\neg P(a) \vee [\neg Q(f(y))]$ is an $\mathcal{I}^-$-all-true conflict clause (all its literals are assigned)

First-order clausal propagation

$C = L_1 \vee \ldots [L_j] \vee \ldots \vee L_k$

- Conflict clause: for all i, $1 \leq i \leq k$, $\mathcal{I}[\Gamma] \models \neg L_i$
- Implied literal and justification:
  for all i, $1 \leq i \neq j \leq k$, $\mathcal{I}[\Gamma] \models \neg L_i$ and $\mathcal{I}[\Gamma] \models L_j$
- All justifications are in the disjoint prefix
- $\mathcal{I}$-all-true clause: either conflict clause or justification

Example I (part III): SGGS finds a refutation

- S contains $\{\ P(a),\ \neg P(x) \vee Q(f(y)),\ \neg P(x) \vee \neg Q(z)\ \}$
- $\Gamma_3 = [P(a)],\ \neg P(a) \vee [Q(f(y))],\ \neg P(a) \vee [\neg Q(f(y))]$
- $\Gamma_4 = [P(a)],\ \neg P(a) \vee [\neg Q(f(y))],\ \neg P(a) \vee [Q(f(y))]$
  by SGGS-move: $\mathcal{I}[\Gamma_4] \models \neg Q(f(y))$
  Conflict: $\neg P(a) \vee [Q(f(y))]$ is a conflict clause
- $\Gamma_5 = [P(a)],\ \neg P(a) \vee [\neg Q(f(y))],\ [\neg P(a)]$ by SGGS-resolution:
  the SGGS-resolvent replaces the non-$\mathcal{I}^-$-all-true parent
- $\Gamma_6 = [\neg P(a)],\ [P(a)],\ \neg P(a) \vee [\neg Q(f(y))]$ by SGGS-move
- $\Gamma_7 = [\neg P(a)],\ \bot,\ \neg P(a) \vee [\neg Q(f(y))]$ by SGGS-resolution

The SGGS inference system I

- Model search: SGGS-extension with literal selection analogue of decision in CDCL
- Conflict solving: if the conflict clause $C[L]$ is $\mathcal{I}$-all-true:
  - SGGS-move it to the left of the clause L is assigned to
  - Analogue of learning and backjumping in CDCL as $C[L]$ enters the disjoint prefix with L as implied literal with justification C

The SGGS inference system II

If the conflict clause $C[L]$ is not $\mathcal{I}$-all-true:

- SGGS-resolve an $\mathcal{I}$-false literal in $C[L]$ with an $\mathcal{I}$-true selected literal in a justification: analogue of explanation in CDCL
- The SGGS-resolvent is still a conflict clause and replaces the parent conflict clause
- SGGS-extension ensures that all $\mathcal{I}$-false literals in $C[L]$ can be resolved away: get either $\perp$ or an $\mathcal{I}$-all-true conflict clause

Example II: SGGS finds a model (part I)

- S contains
  - $P(x,x,a)$, $\quad P(x,y,w) \vee P(y,z,w) \vee \neg P(x,z,w)$
  - $\neg P(x,x,b)$, $\quad P(x,z,w) \vee \neg P(x,y,w) \vee \neg P(y,z,w)$
- $\mathcal{I}$ is $\mathcal{I}^-$ (all-negative)
- $\Gamma_1 = [P(x,x,a)]$
- $\Gamma_2 = [P(x,x,a)],\ P(x,y,a) \vee [P(y,x,a)] \vee \neg P(x,x,a)$
  by SGGS-extension with mgu $\alpha = \{z \leftarrow x, w \leftarrow a\}$
  (selecting $P(x,y,a)$ makes no difference)

Example II (part II)

- S contains
  - $P(x,x,a)$, $P(x,y,w) \vee P(y,z,w) \vee \neg P(x,z,w)$
  - $\neg P(x,x,b)$, $P(x,z,w) \vee \neg P(x,y,w) \vee \neg P(y,z,w)$
- $\Gamma_2 = [P(x,x,a)],\ P(x,y,a) \vee [P(y,x,a)] \vee \neg P(x,x,a)$
  the two selected literals have non-empty intersection
- $\Gamma_3 = [P(x,x,a)],\ P(x,x,a) \vee [P(x,x,a)] \vee \neg P(x,x,a)$,
  $y \not\equiv x \rhd P(x,y,a) \vee [P(y,x,a)] \vee \neg P(x,x,a)$
  by SGGS-splitting to remove the intersection
- SGGS-splitting introduces constraints

Example II (part III)

- S contains
  - $P(x,x,a)$, $\quad P(x,y,w) \vee P(y,z,w) \vee \neg P(x,z,w)$
  - $\neg P(x,x,b)$, $\quad P(x,z,w) \vee \neg P(x,y,w) \vee \neg P(y,z,w)$
- $\Gamma_3 = [P(x,x,a)],\ P(x,x,a) \vee [P(x,x,a)] \vee \neg P(x,x,a),$ $y \not\equiv x \rhd P(x,y,a) \vee [P(y,x,a)] \vee \neg P(x,x,a)$
  the second clause is disposable
- $\Gamma_4 = [P(x,x,a)],\ y \not\equiv x \rhd P(x,y,a) \vee [P(y,x,a)] \vee \neg P(x,x,a)$
  by SGGS-deletion
- $\mathcal{I}[\Gamma_4] \models S$: SGGS halts
- This set of clauses is in EPR

Effectively PRopositional Logic (EPR)

- Also known as the Bernays-Schönfinkel class
- Sentences of the form $\exists^*\forall^*\varphi$
  φ: formula with neither quantifiers nor functions (constants allowed)
- Clausal form: replace $\exists$-quantified variables by Skolem constants; no function symbols; finite Herbrand base; decidable
- But not decidable (directly) by hyperresolution

Recall what is hyperresolution

- Semantic resolution: generate only resolvents false in $\mathcal{I}$
- Hyperresolution: semantic resolution with $\mathcal{I}^-$ or $\mathcal{I}^+$: sign-based semantic guidance
- Positive hyperresolution: resolve a non-positive clause C with as many positive clauses as needed to resolve away with a simultaneous mgu all negative literals in C and get a positive resolvent (false in $\mathcal{I}^-$)
- Negative hyperresolution: dual with $\mathcal{I}^+$

Hyperresolution does not decide EPR (directly)

- S contains
  - $P(x, x, a)$, $P(x, y, w) \vee P(y, z, w) \vee \neg P(x, z, w)$
  - $\neg P(x, x, b)$, $P(x, z, w) \vee \neg P(x, y, w) \vee \neg P(y, z, w)$
- Positive hyperresolution generates infinitely many clauses from $P(x, x, a)$ and $P(x, y, w) \vee P(y, z, w) \vee \neg P(x, z, w)$
- Negative hyperresolution generates infinitely many clauses from $\neg P(x, x, b)$ and $P(x, z, w) \vee \neg P(x, y, w) \vee \neg P(y, z, w)$
- SGGS decides EPR: let's see why

How SGGS makes progress

- Suppose $\bot \notin \Gamma$ and $\mathcal{I}[\Gamma] \not\models S$
- If $\Gamma = dp(\Gamma)$: as $\mathcal{I}[\Gamma] \not\models C$ for some clause $C \in S$, extend Γ hence $\mathcal{I}[\Gamma]$ (SGGS-extension)
- If $\Gamma \neq dp(\Gamma)$: expose intersection (SGGS-splitting) and remove it (SGGS-deletion or SGGS-resolution) or solve conflict (SGGS-resolution, SGGS-splitting, SGGS-move)
- Non-termination may come only from infinitely many SGGS-extensions

Fairness of an SGGS-derivation

- Makes progress whenever $\bot \notin \Gamma$ and $I[\Gamma] \not\models S$
- Applies SGGS-deletion eagerly
- Every SGGS-extension that adds a conflict clause is bundled with conflict solving
- Does not neglect inferences on shorter prefixes to work on longer ones

- Ordering $>^c$ on SGGS trails
- $>^c$ is well-founded on trails of bounded length
- Limit Γ_∞ of a fair derivation: all prefixes stabilize eventually

Fundamental theorems about SGGS

- S: input set of clauses
- A descending chain of length-bounded trails is finite
- A fair derivation is a descending chain
- SGGS is refutationally complete:
  if S is unsatisfiable, SGGS halts with a refutation
- SGGS is model-complete in the limit:
  if S is satisfiable, $\mathcal{I}[\Gamma_\infty] \models S$

Two approaches to get SGGS decision procedures

1. Show that the length of SGGS-trails is bounded
2. Show that if hyperresolution halts so does SGGS

First approach: finite basis

- S: input set of clauses
- $\mathcal{H}$ its Herbrand universe and $\mathcal{A}$ its Herbrand base
- Finite basis: finite subset $\mathcal{B} \subseteq \mathcal{A}$
- Finite set $\mathcal{H}(\mathcal{B}) \subseteq \mathcal{H}$ of the ground subterms of atoms in $\mathcal{B}$
- An SGGS-derivation is in the finite basis $\mathcal{B}$ if all ground instances of all clauses ever appearing on the trail are made of atoms in $\mathcal{B}$

Termination of SGGS in a finite basis

- Input set S and finite basis $\mathcal{B}$
- If a fair SGGS-derivation is in $\mathcal{B}$:
  - At all stages the length of the trail is upper bounded by $|\mathcal{B}|$ ($|\Gamma_j| \leq |\mathcal{B}| + 1$ and $|\Gamma_j| \leq |\mathcal{B}|$ if $dp(\Gamma_j) = \Gamma_j$)
  - The derivation is finite
  - If S is satisfiable then it has a model of cardinality $|\mathcal{H}(\mathcal{B})| + 1$ that can be extracted from the final Γ

Decidability by the finite basis approach

- Fragment $\mathcal{F}$
- Set S of clauses in $\mathcal{F}$
- Show that for all S there exists a finite basis $\mathcal{B}$ (that typically depends on S)
- Then any fair SGGS-strategy is a model-constructing decision procedure for $\mathcal{F}$
- $\mathcal{F}$ has the small model property: every satisfiable S has a model whose cardinality is upper-bounded

SGGS decides the stratified fragment hence EPR

Stratified fragment

- Well-founded ordering $>_s$ on sorts:
  if $f: s_1 \times \ldots \times s_n \rightarrow s$ then $s_i >_s s$
- Sort-dependency graph: arc from s_i to s
- No cycles: no series such as $a, f(a), f^2(a), f^3(a), \ldots$ or $a, f(a), g(f(a)), f(g(f(a))), \ldots$: the Herbrand base is finite
- EPR is the special case with one sort: no function symbols
- Check stratification after Skolemization ($\exists^*\forall^*$ is ok)
- The finite basis $\mathcal{B}$ is the Herbrand base itself

Second approach: ground-preserving clauses

Clause C: C^+ positive literals; C^- negative literals

- Positively ground-preserving: $\mathcal{V}ar(C) \subseteq \mathcal{V}ar(C^-)$
- Negatively ground-preserving: $\mathcal{V}ar(C) \subseteq \mathcal{V}ar(C^+)$
- S positively ground-preserving: positive clauses are ground
- Positive hyperresolution only generates ground clauses
- SGGS:
  - $\mathcal{I}^-$ is suitable for positively ground-preserving set
  - $\mathcal{I}^+$ is suitable for negatively ground-preserving set

Second approach: from hyperresolution to SGGS

- S positively ground-preserving
- SGGS with $\mathcal{I}^-$ only generates ground clauses
- For every clause C that SGGS puts on the trail, C^+ is a subset of a positive hyperresolvent
- If positive hyperresolution halts, so does SGGS with $\mathcal{I}^-$
- SGGS decides the PVD (positive variable dominated) and BDI (bounded depth increase) fragments
- For PVD it can be proved also by the finite basis approach

Negative results with sign-based semantic guidance

SGGS with $\mathcal{I}^-$ or $\mathcal{I}^+$ does not decide the following fragments that admit (ordered, not hyper) resolution-based decision procedures:

- Ackermann ($\exists^*\forall\exists^*\varphi$)
- Monadic (no functions, unary predicates)
- FO^2 (no functions, only 2 variables)
- Guarded (no functions, quantification only in the form $\forall\bar{y}.(R(\bar{x},\bar{y}) \supset \psi[\bar{x},\bar{y}])$ and $\exists\bar{y}.(R(\bar{x},\bar{y}) \wedge \psi[\bar{x},\bar{y}])$)

Can we use SGGS to discover new decidable fragments?

Restrained clauses: intuition

$S = \{\ P(s^{10}(0), s^9(0)),\ \neg P(s(s(x)), y) \vee P(x, s(y)),\ \neg P(s(0), 0)\ \}$
Positively ground-preserving, $\mathcal{I}$ is $\mathcal{I}^-$

- $\Gamma_1 = [P(10,9)]$
- $\Gamma_2 = [P(10,9)],\ \neg P(10,9) \vee [P(8,10)]$
- $\Gamma_3 = [P(10,9)],\ \neg P(10,9) \vee [P(8,10)],\ \neg P(8,10) \vee [P(6,11)]$

  ....
- $\Gamma_6 = [P(10,9)], \ldots\ \neg P(2,13) \vee [P(0,14)]$ and $\mathcal{I}[\Gamma_6] \models S$

$P(s(s(x)), y) \succ P(x, s(y))$
$\succ$: KBO or LPO with $P > s$ in the precedence

Restrained clauses

Restraining quasi-ordering $\succeq$:

- Stable (under substitutions)
- $\succ$ well-founded
- $\approx = \succeq \cap \preceq$ has finite equivalence classes

Clause C is (strictly) positively restrained:

- Positively ground-preserving: $\mathcal{V}ar(C) \subseteq \mathcal{V}ar(C^-)$
- For all non-ground $L \in C^+$ there exists $M \in C^-$ such that $M \succeq L$ ($M \succ L$)

Why a quasi-ordering?
$differ(x, y) \vee \neg differ(y, x)$: $differ(x, y) \approx differ(y, x)$

SGGS decides the restrained fragments

S restrained set of clauses, $\mathcal{A}$ its Herbrand base

- $\mathcal{A}_S$: set of ground atoms in S
- Finite basis: $\mathcal{A}_S^{\preceq} = \{L \,:\, L \in \mathcal{A}, \exists M \in \mathcal{A}_S \text{ s.t. } M \succeq L\}$: the ground atoms upper-bounded by those in S
- Any fair SGGS-derivation with suitable $\mathcal{I}$ is in $\mathcal{A}_S^{\preceq}$
- Any fair SGGS-derivation halts, is a refutation if S is unsatisfiable, and constructs a model if S is satisfiable
- Upper bound on model's cardinality: $|\mathcal{H}(\mathcal{A}_S^{\preceq})| + 1$
- Also PO-resolution and positive hyperresolution halt, but they don't construct models

Sort-restrained clauses: intuition

$S = \{\ P(x, f(b)),\ \neg Q(x, a) \vee Q(a, x),\ \neg P(x, f(y)) \vee Q(x, x) \vee P(x, y)\ \}$
$a\colon s_1 \quad b\colon s_2 \quad f: s_2 \rightarrow s_2 \quad P \subseteq s_1 \times s_2 \quad Q \subseteq s_1 \times s_1$
Neither ground-preserving nor stratified
SGGS with $\mathcal{I}^-$ halts:

- $\Gamma_1 = [P(x, f(b))]$
- $\Gamma_2 = [P(x, f(b))],\ \neg P(x, f(b)) \vee Q(x, x) \vee [P(x, b)]$
- $\mathcal{I}[\Gamma_2] \models S$

- Positively ground-preserving for the cyclic sort s_2
- $P(x, f(y)) \succ P(x, y)$ for $\succ$ any KBO or LPO

Sort-restrained clauses

- Generalize restrained and stratified:
  restrained on sorts with infinite domain
- Sort s with infinite domain: path from a cyclic sort to s
- Restraining quasi-ordering (with the subterm property)
- Clause C is positively sort-restrained:
  - Positively ground-preserving on sorts with infinite domain: $\mathcal{V}ar_s(C) \subseteq \mathcal{V}ar_s(C^-)$
  - For all $L \in C^+$ such that $Gr(L)$ is infinite there exists $M \in C^-$ such that $M \succeq L$

SGGS decides the sort-restrained fragments

- Set of all atoms L in S such that $Gr(L)$ is finite
- Smallest instantiation-closed and $\preceq$-closed superset
- Basis $\mathcal{A}^{\preceq}_{S,\Sigma}$ for S: all ground instances
- This basis is finite
- Any fair SGGS-derivation with suitable $\mathcal{I}$ is in $\mathcal{A}^{\preceq}_{S,\Sigma}$
- Any fair SGGS-derivation halts, is a refutation if S is unsatisfiable, and constructs a model if S is satisfiable
- Upper bound on model's cardinality: $|\mathcal{H}(\mathcal{A}^{\preceq}_{S,\Sigma})| + 1$

Another new decidable fragment

- SGGS decides the sort-refined PVD fragment
  generalize stratified and PVD
  PVD on sorts with infinite domain
- PO-resolution and positive hyperresolution do not decide the sort-restrained and sort-refined PVD fragments
  (at least not directly)

How to determine that a set of clauses is restrained

- Extract from S a rewrite system $\mathcal{R}$ on atoms
- For all clauses $C \in S$, for all non-ground literals $L \in C^+$ there exists literal $\neg M \in C^-$ such that $(M \rightarrow L) \in \mathcal{R}$
- $\rightarrow_{\mathcal{R}}$ terminating: $\rightarrow^*_{\mathcal{R}}$ restraining quasi-ordering
- Add $\mathcal{E}$ for permutations: $differ(x, y) \approx differ(y, x)$
- Rewriting modulo: $\rightarrow_{\mathcal{R}/\mathcal{E}}$ is $\leftrightarrow^*_{\mathcal{E}} \circ \rightarrow_{\mathcal{R}} \circ \leftrightarrow^*_{\mathcal{E}}$
- $\rightarrow_{\mathcal{R}/\mathcal{E}}$ terminating, $\mathcal{V}ar(t) = \mathcal{V}ar(u)$ for all $t \simeq u$ in $\mathcal{E}$, and $\leftrightarrow^*_{\mathcal{E}}$ has finite equivalence classes: $\rightarrow^*_{\mathcal{R}/\mathcal{E}}$ restraining quasi-ordering
- Apply a termination tool such as AProVE or $\mathrm{T_TT_2}$

Experimental results

- Source of clause sets: TPTP 7.4.0
- First-order problems without equality: 5,000 benchmarks
- Size and number of candidate rewrite systems grow exponentially with number of literals in the set of clauses
- 1,539 clause sets yield too big rewrite systems
- AProVE and $\mathsf{T_TT_2}$ applied to at most 100 candidates per set
- Out of the remaining 3,461 problems:
  - 2,137 belong to at least one decidable class
  - 1,399 (66%) belong to at least one SGGS-decidable class
  - 97 are discovered decidable for the first time

The Koala SGGS-based prototype theorem prover

- Written in OCAML by Sarah Winkler
- Trail as list of clauses with constraints in standard form and selected literals in a discrimination tree to compute substitutions for SGGS-extensions
- Fair search plans
- In the experiments: $\mathcal{I}^-$ by default and $\mathcal{I}^+$ if the input is negatively ground-preserving

Experimental results with Koala

Koala solves (Time-out = 300 sec wall-clock time):

- 90% of the new decidable problems
- 78% of the problems in the SGGS-decidable classes
- 58% of the Horn problems
- 43% of the problems whose sat/unsat status is known performing better on sat (64%) than unsat (38%)

Comparison with the state of the art:
in line with E 2.4, Vampire 4.4, and iProver 3.5 in terms of # of problems solved in the new SGGS-decidable classes

Current and future work

- Behavior of SGGS in the Horn case
- More work on strategies and inner algorithms for SGGS
- Further development of the Koala prover
- Extension to equality
  - Integrate SGGS and superposition: SGGS(superposition)
  - Integrate SGGS into CDSAT: CDSAT(SGGS)
- Initial interpretations not based on sign:
  - Satisfiable subset of ground clauses
  - SAT or SMT solver generates a model
  - Use it as initial interpretation for SGGS

References

- Semantically-guided goal-sensitive reasoning: decision procedures and the Koala prover. In preparation, 48 pages (with Sarah Winkler)
- SGGS decision procedures. Proc. 10th IJCAR, Springer, LNAI 12166:356–374, 2020 (with Sarah Winkler)
- Semantically-guided goal-sensitive reasoning: inference system and completeness. *Journal of Automated Reasoning*, 59(2):165–218, 2017 (with David A. Plaisted).
- Semantically-guided goal-sensitive reasoning: model representation. *Journal of Automated Reasoning* 56(2):113–141, 2016 (with David A. Plaisted).
- SGGS theorem proving: an exposition. Proc. 4th PAAR Workshop, EPiC 31:25-38, 2015 (with David A. Plaisted)

Thanks

Thank you!