Producing Conflict Sets for Combinations of Theories

S. Ranise, C. Ringeissen, and D.-K. Tran

[1]LORIA & INRIA-Lorraine
Nancy (France)

[2]Dipartimento di Informatica
Università degli Studi di Milano
Milano (Italia)

Seattle, August 22, 2006


Outline



Outline


Outline

Outline

Outline

Satisfiability Modulo Theory (SMT) Tools

- Efficient (in practice) tools to check the satisfiability of arbitrary Boolean combinations of ground atoms modulo a background theory T
- Based on the integration of a Boolean solver (used to enumerate Boolean assignments) and a decision procedure for the background theory T (used to prune unsatisfiable assignments wrt. T)
- Efficiency obtained (among other techniques) by minimizing the number of calls to the decision procedure for T ... *How?*
- The decision procedure for T is required to compute ***explanations*** of the unsatisfiability of the input set S of ground literals
  - *explanations* of the unsatisfiability of S = "*small*" subset of S which is still unsatisfiable in T

Satisfiability Modulo Theory (SMT) Tools

- Efficient (in practice) tools to check the satisfiability of arbitrary Boolean combinations of ground atoms modulo a background theory T
- Based on the integration of a Boolean solver (used to enumerate Boolean assignments) and a decision procedure for the background theory T (used to prune unsatisfiable assignments wrt. T)
- Efficiency obtained (among other techniques) by minimizing the number of calls to the decision procedure for T ... *How?*
- The decision procedure for T is required to compute ***explanations*** of the unsatisfiability of the input set S of ground literals
  - *explanations* of the unsatisfiability of S = "*small*" subset of S which is still unsatisfiable in T

Satisfiability Modulo Theory (SMT) Tools

- Efficient (in practice) tools to check the satisfiability of arbitrary Boolean combinations of ground atoms modulo a background theory T
- Based on the integration of a Boolean solver (used to enumerate Boolean assignments) and a decision procedure for the background theory T (used to prune unsatisfiable assignments wrt. T)
- Efficiency obtained (among other techniques) by minimizing the number of calls to the decision procedure for T... *How?*
- The decision procedure for T is required to compute ***explanations*** of the unsatisfiability of the input set S of ground literals
  - *explanations* of the unsatisfiability of S = "*small*" subset of S which is still unsatisfiable in T

Satisfiability Modulo Theory (SMT) Tools

- Efficient (in practice) tools to check the satisfiability of arbitrary Boolean combinations of ground atoms modulo a background theory T
- Based on the integration of a Boolean solver (used to enumerate Boolean assignments) and a decision procedure for the background theory T (used to prune unsatisfiable assignments wrt. T)
- Efficiency obtained (among other techniques) by minimizing the number of calls to the decision procedure for T... *How?*
- The decision procedure for T is required to compute ***explanations*** of the unsatisfiability of the input set S of ground literals
  - *explanations* of the unsatisfiability of S = "*small*" subset of S which is still unsatisfiable in T

Computing Conflict Sets

- ***QUESTION***: How to lift decision procedures solving the satisfiability problem in a theory T to also return conflict sets*?*
- ***PARTIAL ANSWER***: Some proposals ***[Fon04, dMRS04, NO05, ST05]*** when T is a single theory (T = "*theory of uninterpreted function symbols*")
- ***STILL UNANSWERED***: how to compute conflict sets in combinations of theories, e.g. $T = T_1 \cup T_2$*?*
- ***REMARK***: all the proposed solutions compute "***small***" but not necessarely ***minimal*** conflict sets
  - *minimal* conflict set CS = there is no $CS' \subset CS$ which is unsatisfiable in T to ensure efficiency... NP-hard for the theory of uninterpreted function symbols*!*

Computing Conflict Sets

- ***QUESTION***: How to lift decision procedures solving the satisfiability problem in a theory T to also return conflict sets*?*
- ***PARTIAL ANSWER***: Some proposals ***[Fon04, dMRS04, NO05, ST05]*** when T is a single theory (T = "*theory of uninterpreted function symbols*")
- ***STILL UNANSWERED***: how to compute conflict sets in combinations of theories, e.g. $T = T_1 \cup T_2$*?*
- ***REMARK***: all the proposed solutions compute "***small***" but not necessarely ***minimal*** conflict sets
  - *minimal* conflict set CS = there is no $CS' \subset CS$ which is unsatisfiable in T to ensure efficiency... NP-hard for the theory of uninterpreted function symbols*!*

Computing Conflict Sets

- ***QUESTION***: How to lift decision procedures solving the satisfiability problem in a theory T to also return conflict sets*?*
- ***PARTIAL ANSWER***: Some proposals ***[Fon04, dMRS04, NO05, ST05]*** when T is a single theory (T = "*theory of uninterpreted function symbols*")
- ***STILL UNANSWERED***: how to compute conflict sets in combinations of theories, e.g. $T = T_1 \cup T_2$*?*
- ***REMARK***: all the proposed solutions compute "***small***" but not necessarely ***minimal*** conflict sets
  - *minimal* conflict set CS = there is no $CS' \subset CS$ which is unsatisfiable in T to ensure efficiency... NP-hard for the theory of uninterpreted function symbols*!*

Computing Conflict Sets

- ***QUESTION***: How to lift decision procedures solving the satisfiability problem in a theory T to also return conflict sets*?*
- ***PARTIAL ANSWER***: Some proposals ***[Fon04, dMRS04, NO05, ST05]*** when T is a single theory (T = "*theory of uninterpreted function symbols*")
- ***STILL UNANSWERED***: how to compute conflict sets in combinations of theories, e.g. $T = T_1 \cup T_2$*?*
- ***REMARK***: all the proposed solutions compute "***small***" but not necessarely ***minimal*** conflict sets
  - *minimal* conflict set CS = there is no $CS' \subset CS$ which is unsatisfiable in T to ensure efficiency... NP-hard for the theory of uninterpreted function symbols*!*

Two contributions

1. Modular computation of conflict sets in combinations of theories by extending the Nelson-Oppen combination schema ***[NO79]*** via the concept of ***explanation graphs***

2. Study of the relationship between the computed “small” conflict sets and minimal ones via the concept of ***quasi-conflict set***

Two contributions

1. Modular computation of conflict sets in combinations of theories by extending the Nelson-Oppen combination schema ***[NO79]*** via the concept of ***explanation graphs***

2. Study of the relationship between the computed “small” conflict sets and minimal ones via the concept of ***quasi-conflict set***

Explanation graph

- Intuition: a labelled graph (V, E) where
  - V is a set of constants (occurring in the input set of literals)
  - E is an elementary equality (i.e., an equality between constants)
  - labels are explanations of elementary equalities
- An explanation EX for an elementary equality $x = y$ is such that
  - EX is satisfiable in the background theory T
  - $T \cup EX \models x = y$
- An explanation EX for an elementary equality $x = y$ is minimal iff there is no $EX' \subset EX$ which is an explanation for $x = y$

Property

An explanation EX is minimal for $x = y$ iff $EX \cup \{x \neq y\}$ is a minimal conflict set.

Explanation graph

- Intuition: a labelled graph (V, E) where
  - V is a set of constants (occurring in the input set of literals)
  - E is an elementary equality (i.e., an equality between constants)
  - labels are explanations of elementary equalities
- An explanation EX for an elementary equality $x = y$ is such that
  - EX is satisfiable in the background theory T
  - $T \cup EX \models x = y$
- An explanation EX for an elementary equality $x = y$ is minimal iff there is no $EX' \subset EX$ which is an explanation for $x = y$

Property

An explanation EX is minimal for $x = y$ iff $EX \cup \{x \neq y\}$ is a minimal conflict set.

Explanation graph

- Intuition: a labelled graph (V, E) where
  - V is a set of constants (occurring in the input set of literals)
  - E is an elementary equality (i.e., an equality between constants)
  - labels are explanations of elementary equalities
- An explanation EX for an elementary equality $x = y$ is such that
  - EX is satisfiable in the background theory T
  - $T \cup EX \models x = y$
- An explanation EX for an elementary equality $x = y$ is minimal iff there is no $EX' \subset EX$ which is an explanation for $x = y$

Property

An explanation EX is minimal for $x = y$ iff $EX \cup \{x \neq y\}$ is a minimal conflict set.

An explanation graph at work: congruence closure

Init

$$\Omega; E; G \vdash$$
$$\Omega; E; \mathit{Insert}(G, z = z', \{z = t, z' = t\})$$
$$\text{if } \begin{cases} z = t, z' = t \in \Omega \\ z \neq z', (z, z') \notin CP(G) \end{cases}$$

Ins

$$\Omega; E \cup \{x = x'\}; G \vdash$$
$$\Omega; E; \mathit{Insert}(G, x = x', \{x = x'\})$$
$$\text{if } x \neq x', (x, x') \notin CP(G)$$

Skip

$$\Omega; E \cup \{x = x'\}; G \vdash$$
$$\Omega; E; G$$
$$\text{if } x = x' \text{ or } (x, x') \in CP(G)$$

An explanation graph at work: congruence closure (cont'd)

Cong $\Omega; E; G \vdash$
$\Omega; E; \mathit{Insert}(G, z = z', \{z = t, z' = t'\} \cup \bigcup_{j \in J} \{y_j = y'_j\})$

$$\text{if} \begin{cases} z = t \equiv f(y_1, \ldots, y_n), z' = t' \equiv f(y'_1, \ldots, y'_n) \in \Omega \\ z \neq z', (z, z') \notin CP(G) \\ I, J \text{ is a partition of } \{1, \ldots, n\} \text{ such that } J \neq \emptyset \text{ and} \\ (\forall i \in I : y_i = y'_i), (\forall j \in J : (y_j, y'_j) \in CP(G)) \end{cases}$$

Similar to the approach (based on proof forests) in ***[NO05]***...

Nelson-Oppen schema *[NO79]* with explanation graphs

- ***GOAL***: compute conflict sets for $T_1 \cup T_2$ when
  - the theory T_i is convex, stably-infinite, for which a satisfiability procedure is available
  - T_1 and T_2 are signature-disjoint
- ***Intuition***: the Nelson-Oppen method consists in exchanging entailed elementary equalities between the two procedures until...
- ***Key idea***: the unsatisfiability in $T_1 \cup T_2$ can be explained according to two kinds of explanations:
  1. the explanation of entailed elementary equalities
  2. the explanation of the unsatisfiability in a component theory

  To combine these explanations, the key idea is to use specialized satisfiability procedures (called ***explanation engines***) with the capability of generating explanation graphs in order to store entailed equalities

Nelson-Oppen schema *[NO79]* with explanation graphs

- ***GOAL***: compute conflict sets for $T_1 \cup T_2$ when
  - the theory T_i is convex, stably-infinite, for which a satisfiability procedure is available
  - T_1 and T_2 are signature-disjoint
- ***Intuition***: the Nelson-Oppen method consists in exchanging entailed elementary equalities between the two procedures until...
- ***Key idea***: the unsatisfiability in $T_1 \cup T_2$ can be explained according to two kinds of explanations:
  1. the explanation of entailed elementary equalities
  2. the explanation of the unsatisfiability in a component theory

  To combine these explanations, the key idea is to use specialized satisfiability procedures (called ***explanation engines***) with the capability of generating explanation graphs in order to store entailed equalities

Nelson-Oppen schema *[NO79]* with explanation graphs

- ***GOAL***: compute conflict sets for $T_1 \cup T_2$ when
  - the theory T_i is convex, stably-infinite, for which a satisfiability procedure is available
  - T_1 and T_2 are signature-disjoint
- ***Intuition***: the Nelson-Oppen method consists in exchanging entailed elementary equalities between the two procedures until...
- ***Key idea***: the unsatisfiability in $T_1 \cup T_2$ can be explained according to two kinds of explanations:
  1. the explanation of entailed elementary equalities
  2. the explanation of the unsatisfiability in a component theory

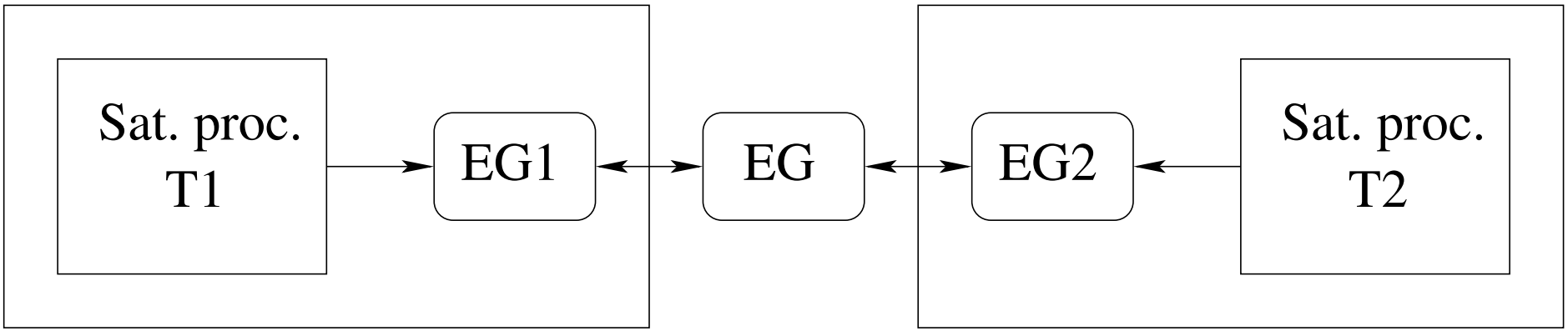
  To combine these explanations, the key idea is to use specialized satisfiability procedures (called ***explanation engines***) with the capability of generating explanation graphs in order to store entailed equalities

Nelson-Oppen schema *[NO79]* with explanation graphs

- ***GOAL***: compute conflict sets for $T_1 \cup T_2$ when
  - the theory T_i is convex, stably-infinite, for which a satisfiability procedure is available
  - T_1 and T_2 are signature-disjoint
- ***Intuition***: the Nelson-Oppen method consists in exchanging entailed elementary equalities between the two procedures until...
- ***Key idea***: the unsatisfiability in $T_1 \cup T_2$ can be explained according to two kinds of explanations:
  1. the explanation of entailed elementary equalities
  2. the explanation of the unsatisfiability in a component theory

  To combine these explanations, the key idea is to use specialized satisfiability procedures (called ***explanation engines***) with the capability of generating explanation graphs in order to store entailed equalities

Modular combination of explanation engines: idea



Modular combination of explanation engines: formally

Explanation engine $\mu EX(\Omega, E) = (\Omega', E', G)$...

$\text{Unsat}_{=1}$

$$\Omega_1; \Delta_V; G; \Omega_2 \vdash$$
$$\mathit{false}\{(\Omega'_1, E'_1, G')\}$$
$$\text{if} \begin{cases} \mu EX_1(\Omega_1, Eq(G)) = (\Omega'_1, E'_1, G_1) \;\&\; \Omega'_1 \neq \emptyset \\ G' = \mathit{Merge}(G, G_1) \end{cases}$$

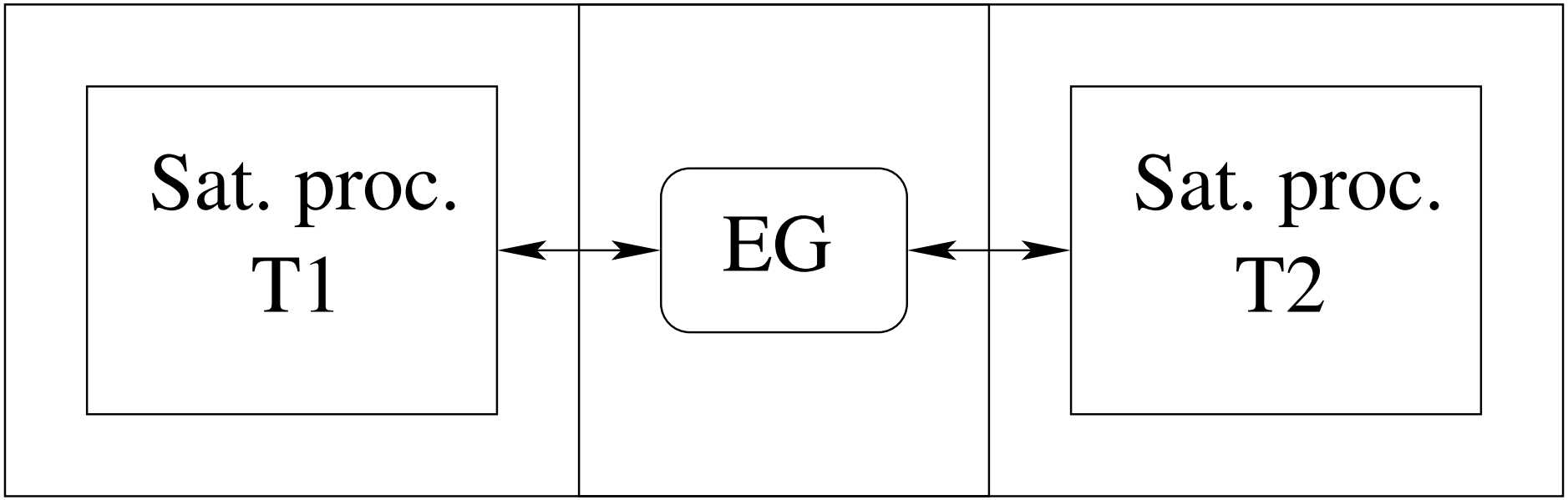
$\text{Unsat}_{\neq}$

$$\Omega_1; \Delta_V; G; \Omega_2 \vdash$$
$$\mathit{false}\{(\{x \neq y\}, \{x = y\}, G)\}$$
$$\text{if } (x, y) \in CP(G) \text{ and } x \neq y \in \Delta_V$$

Deduction_1

$$\Omega_1; \Delta_V; G; \Omega_2 \vdash$$
$$\Omega_1; \Delta_V; G'; \Omega_2$$
$$\text{if} \begin{cases} \mu EX_1(\Omega_1, Eq(G)) = (\emptyset, E_1, G_1) \\ G' = \mathit{Merge}(G, G_1) \\ G' \neq G \end{cases}$$

Modular combination of explanation engines: refinement



Quasi-conflict sets: informally

- The tuple (ψ, E, G) computed by the previous schema is such that
  - $\psi \cup E$ is unsatifiable,
  - ψ is a satisfiable subset of the input set, and
  - E is a set of (entailed) elementary equalities explained in G
- $\psi \cup E$ is not a conflict set of the input set φ since it may contain literals which are not in φ
- However, it is easy to extract a "true" conflict set from $\psi \cup E$ since E is entailed by φ and the related explanations are encoded in the associated explanation graph G
- We define the tuple (ψ, E, G) a ***quasi-conflict set*** and it is possible to define an ***ordering*** on such tuples which allows us to introduce the ***minimality of quasi-conflict sets***

Quasi-conflict sets: informally

- The tuple (ψ, E, G) computed by the previous schema is such that
  - $\psi \cup E$ is unsatifiable,
  - ψ is a satisfiable subset of the input set, and
  - E is a set of (entailed) elementary equalities explained in G
- $\psi \cup E$ is not a conflict set of the input set φ since it may contain literals which are not in φ
- However, it is easy to extract a "true" conflict set from $\psi \cup E$ since E is entailed by φ and the related explanations are encoded in the associated explanation graph G
- We define the tuple (ψ, E, G) a ***quasi-conflict set*** and it is possible to define an ***ordering*** on such tuples which allows us to introduce the ***minimality of quasi-conflict sets***

Quasi-conflict sets: basic properties

Property

If (ψ, E, G) *is a quasi-conflict set of the input set* φ *of literals, then* $\psi \cup Lit(G)$ *is a conflict set of* φ*.*

Theorem

Let (ψ, E, G) *be a quasi-conflict set of* φ *such that* $\psi \cup E$ *is a minimal conflict set. If all edges of* $G_{|E}$ *are minimally explained then* $(\psi, E, G_{|E})$ *is a minimal quasi-conflict set of* φ*.*

Discussion

- Two contributions
  1. Modular computation of conflict sets in combinations of theories by combining ***explanation engines*** (extension of the Nelson-Oppen combination schema)
  2. ***Quasi-conflict sets*** as a characterization of the minimality that can be obtained in practice when combining satisfiability procedures
- Delayed Theory Combination ***[BBC+05]*** as an alternative to avoid the computation of conflict sets in combinations of theories...

Discussion

- Two contributions
  1. Modular computation of conflict sets in combinations of theories by combining ***explanation engines*** (extension of the Nelson-Oppen combination schema)
  2. ***Quasi-conflict sets*** as a characterization of the minimality that can be obtained in practice when combining satisfiability procedures
- Delayed Theory Combination ***[BBC+05]*** as an alternative to avoid the computation of conflict sets in combinations of theories...

Marco Bozzano, Roberto Bruttomesso, Alessandro Cimatti, Tommi A. Junttila, Silvio Ranise, Peter van Rossum, and Roberto Sebastiani.
Efficient satisfiability modulo theories via delayed theory combination.
In *Proc. of Computer Aided Verification, 17th International Conference, CAV 2005, Edinburgh, Scotland, UK, July 6-10*, pages 335–349, 2005.

Leonardo de Moura, Harald Rueß, and Natarajan Shankar.
Justifying Equality.
In *Proc. of the Workshop of the Proceedings of the Pragmatics of Decision Procedures for Automated Reasoning (PDPAR'04)*, 2004.

Pascal Fontaine.
Techniques for Verification of Concurrent Systems with Invariants.
PhD thesis, Université de Liège, 2004.

Greg Nelson and Derek C. Oppen.


Simplification by cooperating decision procedures.
ACM Trans. on Progr. Lang. and Sys., 1(2):245–257, 1979.

Robert Nieuwenhuis and Albert Oliveras.
Proof-Producing Congruence Closure.
In *Proc. of the 16th Int. Conf. on Rewriting Techniques and Applications (RTA)*, volume 3467 of *LNCS*, pages 453–468, 2005.

A. Stump and L.-Y. Tang.
The Algebra of Equality Proofs.
In *Proc. of the 16th Int. Conf. on Rewriting Techniques and Applications (RTA)*, volume 3467 of *LNCS*, pages 469–483, 2005.