Satisfiability Modulo Theories and Assignments

Maria Paola Bonacina, Stéphane Graham-Lengrand,
and Natarajan Shankar

Uni. degli Studi di Verona - CNRS - SRI International

CADE, 8th August 2017

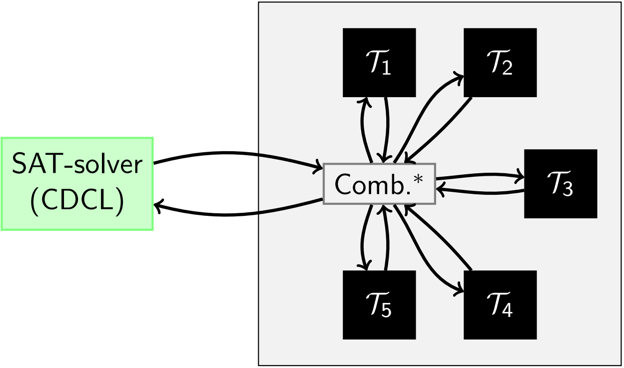
This talk is about the quantifier-free core of SMT-solving.
It involves

- extending CDCL (Conflict-Driven Clause Learning)
- combining theories

This talk is about the quantifier-free core of SMT-solving.
It involves

- extending CDCL (Conflict-Driven Clause Learning)
- combining theories

You may have seen the DPLL($\mathcal{T}$) framework:

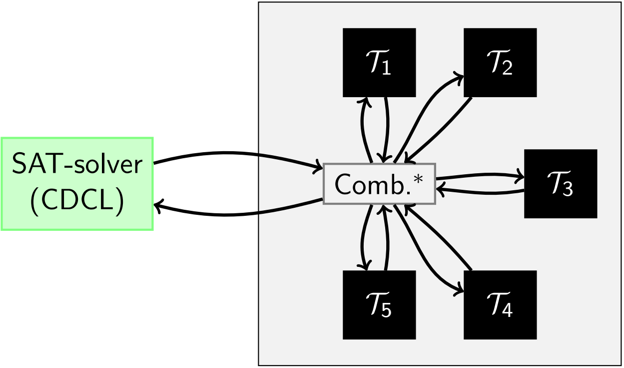


* e.g. equality sharing / Nelson-Oppen [NO79]

This talk is about the quantifier-free core of SMT-solving.
It involves

- extending CDCL (Conflict-Driven Clause Learning)
- combining theories

You may have seen the DPLL($\mathcal{T}$) framework:



* e.g. equality sharing / Nelson-Oppen [NO79]

The material presented here departs from this picture.
Motivation: **conflict-driven reasoning**

Combining conflict-driven reasoning mechanisms

The CDSAT framework

Termination, Soundness and Completeness

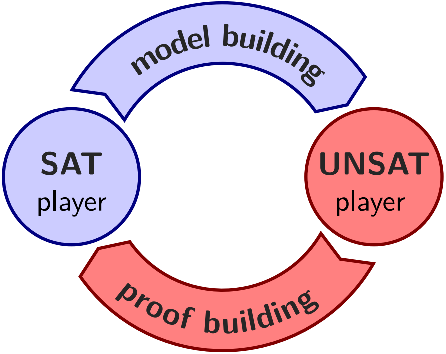
1. Combining conflict-driven reasoning mechanisms

Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

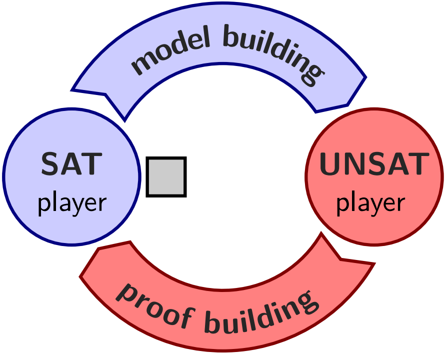


Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

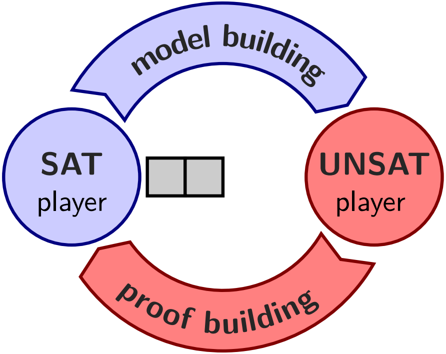


Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

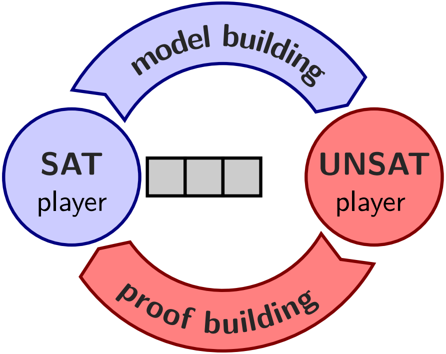


Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

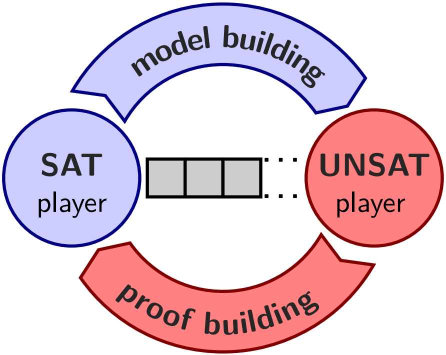


Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

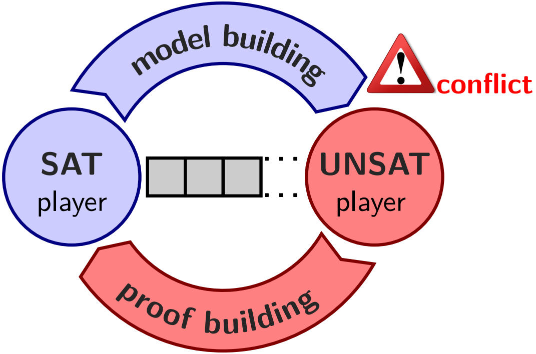


Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.



Conflict-driven reasoning

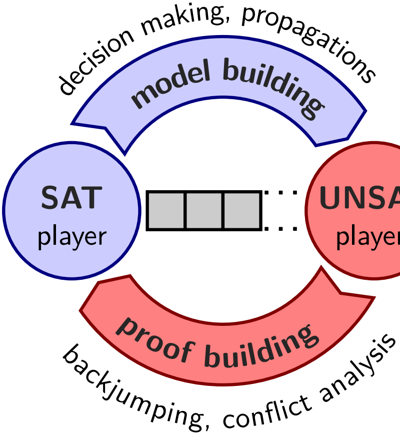
2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

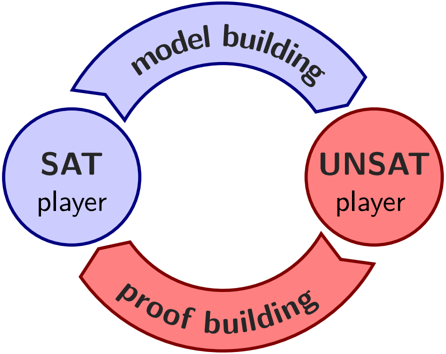
It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified

$a \Rightarrow b$
$b \Rightarrow \overline{a}$
$\overline{a} \Rightarrow \overline{b}$
$\overline{b} \Rightarrow a$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

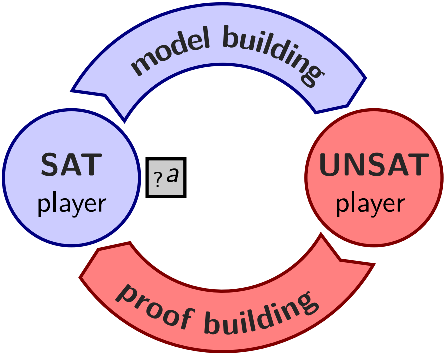
It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified

$$a \Rightarrow b$$
$$b \Rightarrow \overline{a}$$
$$\overline{a} \Rightarrow \overline{b}$$
$$\overline{b} \Rightarrow a$$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

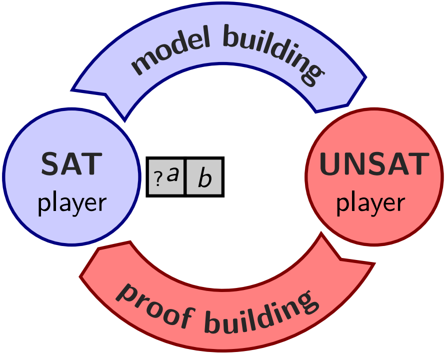
a conflict occurs when a clause is falsified

$a \Rightarrow b$

$b \Rightarrow \overline{a}$

$\overline{a} \Rightarrow \overline{b}$

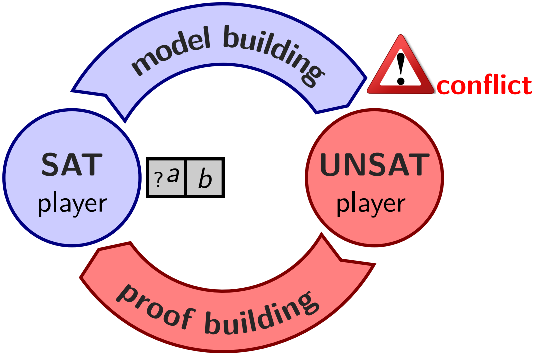
$\overline{b} \Rightarrow a$



Conflict-driven reasoning

- 2-player game to determine whether a problem is sat.
- It involves a trail where a putative model is being described.
- It relies on a notion of conflict between the putative model and the constraints it should satisfy.
- Archetype of conflict-driven reasoning: **CDCL**
- a conflict occurs when a clause is falsified

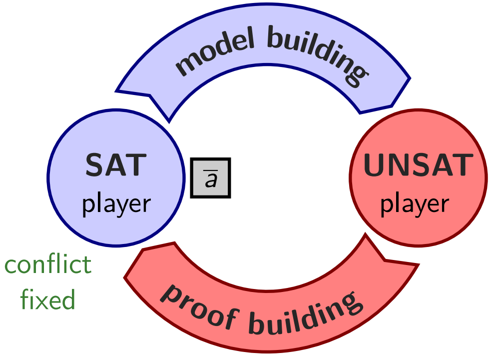
$a \Rightarrow b$
$b \Rightarrow \overline{a}$
$\overline{a} \Rightarrow \overline{b}$
$\overline{b} \Rightarrow a$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.
It involves a trail where a putative model is being described.
It relies on a notion of conflict between the putative model and the constraints it should satisfy.
Archetype of conflict-driven reasoning: **CDCL**
a conflict occurs when a clause is falsified

$a \Rightarrow b$
$b \Rightarrow \overline{a}$
$\overline{a} \Rightarrow \overline{b}$
$\overline{b} \Rightarrow a$
$\overline{a}$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

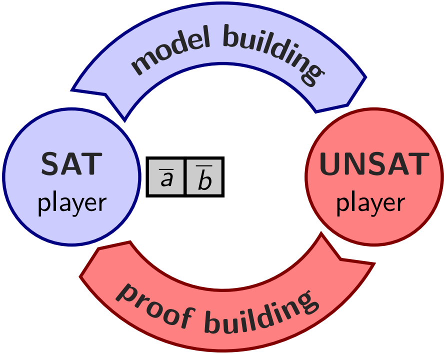
It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified

$a \Rightarrow b$
$b \Rightarrow \overline{a}$
$\overline{a} \Rightarrow \overline{b}$
$\overline{b} \Rightarrow a$
$\overline{a}$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

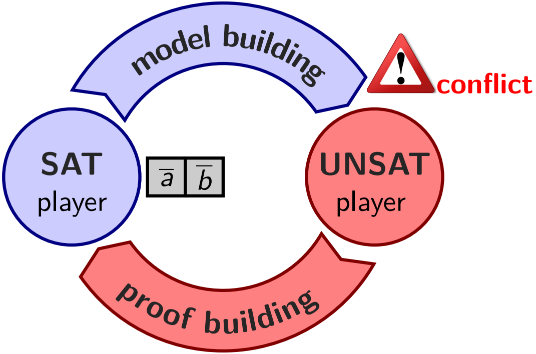
It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified

$a \Rightarrow b$
$b \Rightarrow \overline{a}$
$\overline{a} \Rightarrow \overline{b}$
$\overline{b} \Rightarrow a$
$\overline{a}$



Conflict-driven reasoning

2-player game to determine whether a problem is sat.

It involves a trail where a putative model is being described.

It relies on a notion of conflict between the putative model and the constraints it should satisfy.

Archetype of conflict-driven reasoning: **CDCL**

a conflict occurs when a clause is falsified

$a \Rightarrow b$

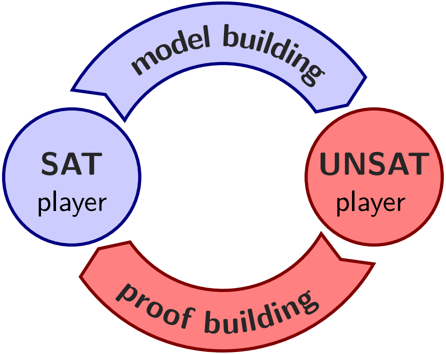
$b \Rightarrow \overline{a}$

$\overline{a} \Rightarrow \overline{b}$

$\overline{b} \Rightarrow a$

$\overline{a}$

$\bot$



Conflict-driven reasoning can be used for (other) theories

Examples:

- LPSAT [WW99]
- Separation logic [WIGG05]
- Linear Rational Arithmetic [MKS09, KTV09, Cot10]
- Linear Integer Arithmetic [Jd11]
- Non-Linear Arithmetic [JdM12]

Conflict-driven reasoning can be used for (other) theories

Examples:

- LPSAT [WW99]
- Separation logic [WIGG05]
- Linear Rational Arithmetic [MKS09, KTV09, Cot10]
- Linear Integer Arithmetic [Jd11]
- Non-Linear Arithmetic [JdM12]

These conflict-driven decision procedures for $\mathcal{T}$-satisfiability

- use assignments to first-order variables (e.g. $x \leftarrow 3/4$) like CDCL uses Boolean assignments to Boolean variables;
- may explain conflicts by introducing atoms that are not in the input.

An example in Linear Rational Arithmetic

$l_0 : (-2\cdot x - y < 0), \qquad l_1 : (x + y < 0), \qquad l_2 : (x < -1)$

unsatisfiable in LRA. Here's how it could start:

An example in Linear Rational Arithmetic

$$l_0 : (-2{\cdot}x - y < 0), \qquad l_1 : (x + y < 0), \qquad l_2 : (x < -1)$$

unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0), \qquad l_1 : (x + y < 0), \qquad l_2 : (x < -1)$

unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, range of possible values for x is empty
  What to do? just undo $y \leftarrow 0$ and remember that $y \neq 0$?

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, range of possible values for x is empty
  What to do? just undo $y \leftarrow 0$ and remember that $y \neq 0$?
- **No!** Clash of bounds suggests a better conflict explanation,
  by inferring $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  It rules out $y \leftarrow 0$,
  but also many values that would fail for the same reasons.

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0), \qquad l_1 : (x + y < 0), \qquad l_2 : (x < -1)$

unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, range of possible values for x is empty
  What to do? just undo $y \leftarrow 0$ and remember that $y \neq 0$?
- **No!** Clash of bounds suggests a better conflict explanation,
  by inferring $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  It rules out $y \leftarrow 0$,
  but also many values that would fail for the same reasons.
- Now undo the guess but keep l_3.

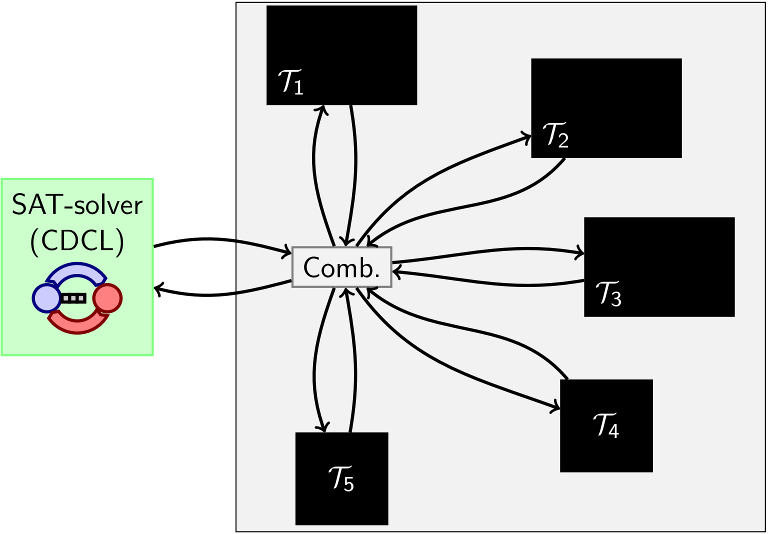
An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

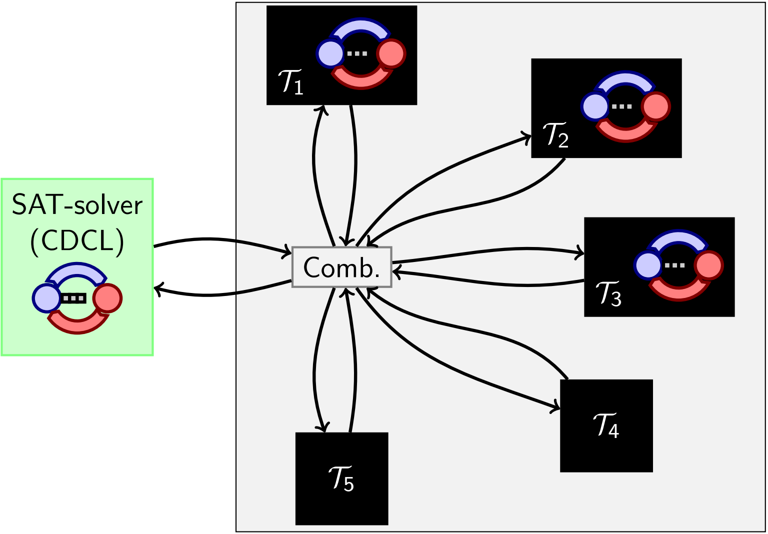
unsatisfiable in LRA. Here's how it could start:

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, range of possible values for x is empty
  What to do? just undo $y \leftarrow 0$ and remember that $y \neq 0$?
- **No!** Clash of bounds suggests a better conflict explanation,
  by inferring $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  It rules out $y \leftarrow 0$,
  but also many values that would fail for the same reasons.
- Now undo the guess but keep l_3.
- and so on...
  (when there is no guess to undo, problem is UNSAT)

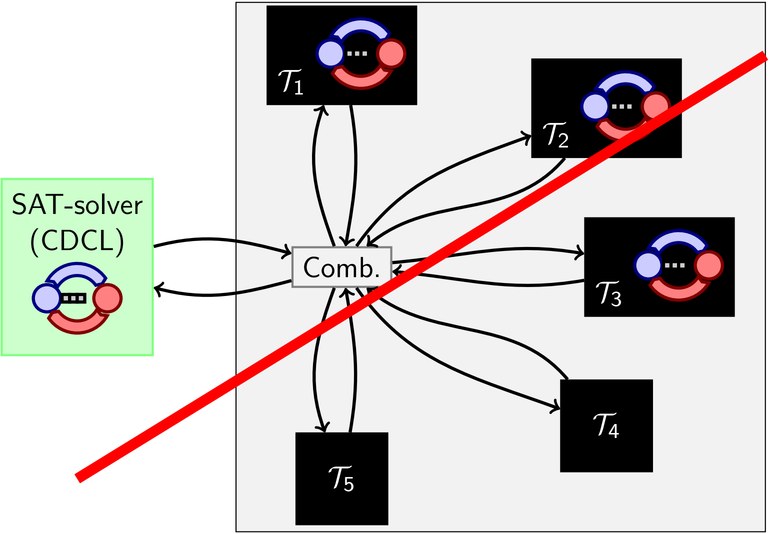
Using conflict-driven reasoning in the traditional scheme?



Using conflict-driven reasoning in the traditional scheme?



Using conflict-driven reasoning in the traditional scheme?



Missing out on tighter integration possibilities,
which overcome some limitations of the DPLL($\mathcal{T}$) interfaces

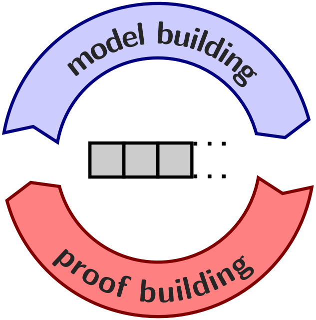
A recent approach: MCSAT (Model-Constructing Sat.)

MCSAT, introduced in [dMJ13, JBdM13],

- departs from the DPLL($\mathcal{T}$) architecture
- organises some combinations into a single conflict-driven loop:

Trail contains

- Boolean assignments
  $a \leftarrow \text{true}$
- First-order assignments
  $y \leftarrow 3/4$



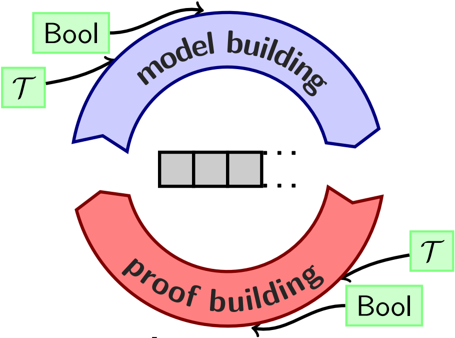
A recent approach: MCSAT (Model-Constructing Sat.)

MCSAT, introduced in [dMJ13, JBdM13],

- departs from the DPLL($\mathcal{T}$) architecture
- organises some combinations into a single conflict-driven loop:

Trail contains

- Boolean assignments
  $a \leftarrow \text{true}$
- First-order assignments
  $y \leftarrow 3/4$



"Some combinations":

- Boolean theory
  $+$ 1 generic theory $\mathcal{T}$ [dMJ13, Jov17]

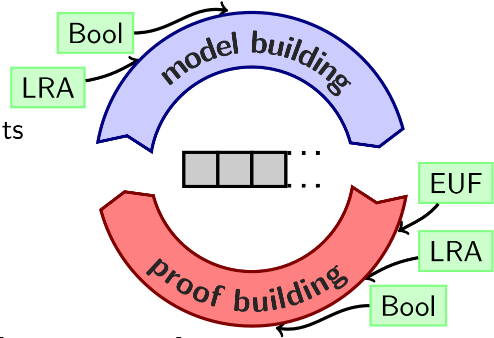
A recent approach: MCSAT (Model-Constructing Sat.)

MCSAT, introduced in [dMJ13, JBdM13],

- departs from the DPLL($\mathcal{T}$) architecture
- organises some combinations into a single conflict-driven loop:

Trail contains

- Boolean assignments
  $a \leftarrow \text{true}$
- First-order assignments
  $y \leftarrow 3/4$



"Some combinations":

- Boolean theory
  $+$ 1 generic theory $\mathcal{T}$ [dMJ13, Jov17]
- Boolean theory $+$ Linear Rational Arithmetic (LRA)
  $+$ Equality with Uninterpreted Functions (EUF) [JBdM13]

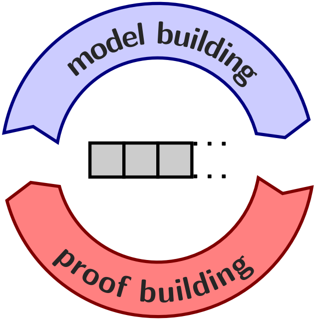
A recent approach: MCSAT (Model-Constructing Sat.)

MCSAT, introduced in [dMJ13, JBdM13],

- departs from the DPLL($\mathcal{T}$) architecture
- organises some combinations into a single conflict-driven loop:

Trail contains

- Boolean assignments
  $a \leftarrow \text{true}$
- First-order assignments
  $y \leftarrow 3/4$



"Some combinations":

- Boolean theory
  $+$ 1 generic theory $\mathcal{T}$ [dMJ13, Jov17]
- Boolean theory $+$ Linear Rational Arithmetic (LRA)
  $+$ Equality with Uninterpreted Functions (EUF) [JBdM13]

Other MCSAT contributions: bit-vectors [ZWR16, GLJ17]

Features of model-constructing satisfiability

- Boolean theory can have the same status as other theories.
- Natively overcomes some limitations of the (basic) DPLL($\mathcal{T}$) interfaces:
  - in order to explain conflicts, terms and literals are exchanged that do not belong to the original problem, providing in some cases exponential speed-ups
    (already the case in some extensions of DPLL($\mathcal{T}$) - see *Splitting on demand* [BNOT06]);

Features of model-constructing satisfiability

- Boolean theory can have the same status as other theories.
- Natively overcomes some limitations of the (basic) DPLL($\mathcal{T}$) interfaces:
  - in order to explain conflicts, terms and literals are exchanged that do not belong to the original problem, providing in some cases exponential speed-ups
    (already the case in some extensions of DPLL($\mathcal{T}$) - see *Splitting on demand* [BNOT06]);
  - determining the truth-value of a literal can be done by evaluation (when its variables are assigned values on the trail);

Features of model-constructing satisfiability

- Boolean theory can have the same status as other theories.
- Natively overcomes some limitations of the (basic) DPLL($\mathcal{T}$) interfaces:
  - in order to explain conflicts, terms and literals are exchanged that do not belong to the original problem, providing in some cases exponential speed-ups
    (already the case in some extensions of DPLL($\mathcal{T}$) - see *Splitting on demand* [BNOT06]);
  - determining the truth-value of a literal can be done by evaluation (when its variables are assigned values on the trail);
  - communicating entailed equalities like $t_1 \simeq t_2$ may be subsumed by the fact that the putative partial model written on the trail determines this equality evaluates to true;

Features of model-constructing satisfiability

- Boolean theory can have the same status as other theories.
- Natively overcomes some limitations of the (basic) DPLL($\mathcal{T}$) interfaces:
  - in order to explain conflicts, terms and literals are exchanged that do not belong to the original problem, providing in some cases exponential speed-ups
    (already the case in some extensions of DPLL($\mathcal{T}$) -
    see *Splitting on demand* [BNOT06]);
  - determining the truth-value of a literal can be done by evaluation (when its variables are assigned values on the trail);
  - communicating entailed equalities like $t_1 \simeq t_2$ may be subsumed by the fact that the putative partial model written on the trail determines this equality evaluates to true;
  - when a theory $\mathcal{T}$ has to decide a value for an assignment, its choice may be informed by inspecting what assignments other theories have written on the trail.

Model-constructing sat. / Conflict-driven reasoning

I reserve *Model-Constructing satisfiability* for the instances of conflict-driven reasoning where theories have canonical models:
If a formula is not valid, a counter-example can be built in that model.
e.g. Boolean logic, integer arithmetic, real arithmetic, bitvectors. . .

Model-constructing sat. / Conflict-driven reasoning

I reserve *Model-Constructing satisfiability* for the instances of conflict-driven reasoning where theories have canonical models:
If a formula is not valid, a counter-example can be built in that model.
e.g. Boolean logic, integer arithmetic, real arithmetic, bitvectors...

- Interpretation of sorts is fixed and known in advance (no cardinality issues);
- Symbols are either interpreted or uninterpreted.

Model-constructing sat. / Conflict-driven reasoning

I reserve *Model-Constructing satisfiability* for the instances of conflict-driven reasoning where theories have canonical models:
If a formula is not valid, a counter-example can be built in that model.
e.g. Boolean logic, integer arithmetic, real arithmetic, bitvectors...

- Interpretation of sorts is fixed and known in advance (no cardinality issues);
- Symbols are either interpreted or uninterpreted.

Left to be determined:

the interpretation of variables and uninterpreted symbols.

This leaves open the following questions

- Specific combinations of MCSAT theories seem simple…
  …once we know how all sorts are interpreted,
  and for each sort there is a clear theory that "owns" it
  (i.e. is in charge of proposing assignments in that sort)

This leaves open the following questions

- Specific combinations of MCSAT theories seem simple. . .
  . . . once we know how all sorts are interpreted,
  and for each sort there is a clear theory that “owns” it
  (i.e. is in charge of proposing assignments in that sort)
- What about the generic combination of n MCSAT theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$?
  What do we need to know about them?
  i.e. what requirements can we enforce to ensure soundness, completeness, and termination of their combination?

This leaves open the following questions

- Specific combinations of MCSAT theories seem simple...
  ...once we know how all sorts are interpreted,
  and for each sort there is a clear theory that "owns" it
  (i.e. is in charge of proposing assignments in that sort)
- What about the generic combination of n MCSAT theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$?
  What do we need to know about them?
  i.e. what requirements can we enforce to ensure soundness, completeness, and termination of their combination?
- What about the generic combination of n theories in general?
  (e.g. it is not clear which sorts they "own",
  they may not have a canonical model, etc)

This leaves open the following questions

- Specific combinations of MCSAT theories seem simple...
  ...once we know how all sorts are interpreted,
  and for each sort there is a clear theory that "owns" it
  (i.e. is in charge of proposing assignments in that sort)
- What about the generic combination of n MCSAT theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$?
  What do we need to know about them?
  i.e. what requirements can we enforce to ensure soundness, completeness, and termination of their combination?
- What about the generic combination of n theories in general?
  (e.g. it is not clear which sorts they "own",
  they may not have a canonical model, etc)
  In particular, what about theories for which we have a black box fit for the equality-sharing / Nelson-Oppen scheme?

This leaves open the following questions

- Specific combinations of MCSAT theories seem simple...
  ...once we know how all sorts are interpreted,
  and for each sort there is a clear theory that "owns" it
  (i.e. is in charge of proposing assignments in that sort)
- What about the generic combination of n MCSAT theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$?
  What do we need to know about them?
  i.e. what requirements can we enforce to ensure soundness, completeness, and termination of their combination?
- What about the generic combination of n theories in general?
  (e.g. it is not clear which sorts they "own",
  they may not have a canonical model, etc)
  In particular, what about theories for which we have a black box fit for the equality-sharing / Nelson-Oppen scheme?
  Is there a way to integrate or generalize both MCSAT and the equality sharing scheme?

The answer: CDSAT

We answer these questions in a framework called **CDSAT** for Conflict-Driven Satisfiability.

- CDSAT generalises conflict-driven reasoning to generic combinations of disjoint theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$
- CDSAT solves the problem of combining multiple conflict-driven $\mathcal{T}_k$-satisfiability procedures into a conflict-driven $(\bigcup_{k=1}^{n} \mathcal{T}_k)$-satisfiability procedure
- CDSAT reduces to MCSAT when it combines Boolean reasoning with 1 MCSAT-procedure
- CDSAT can integrate black-box procedures, and reduces to the equality-sharing scheme if only such procedures are used

The answer: CDSAT

We answer these questions in a framework called **CDSAT** for Conflict-Driven Satisfiability.

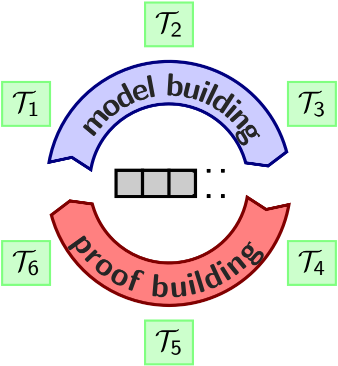
- CDSAT generalises conflict-driven reasoning to generic combinations of disjoint theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$
- CDSAT solves the problem of combining multiple conflict-driven $\mathcal{T}_k$-satisfiability procedures into a conflict-driven $(\bigcup_{k=1}^{n} \mathcal{T}_k)$-satisfiability procedure
- CDSAT reduces to MCSAT when it combines Boolean reasoning with 1 MCSAT-procedure
- CDSAT can integrate black-box procedures, and reduces to the equality-sharing scheme if only such procedures are used

We identify sufficient requirements on theory reasoning modules for the combined system to be sound, complete, and terminating.

2. The CDSAT framework

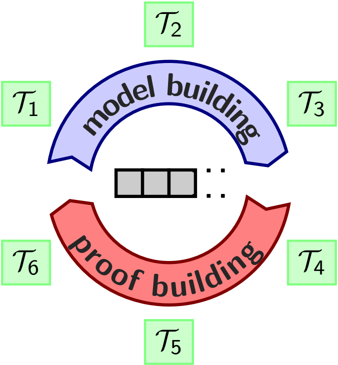
The global picture

... is roughly the same as before (all theories somehow participate to the main conflict-driven loop):



The global picture

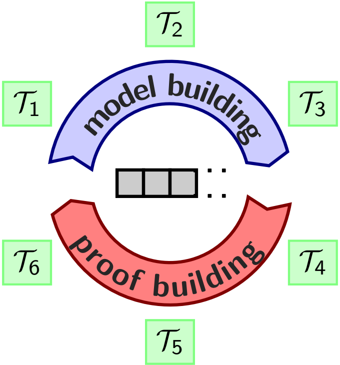
...is roughly the same as before (all theories somehow participate to the main conflict-driven loop):



...except that it it now **parametric** in $\mathcal{T}_1, \ldots, \mathcal{T}_n$.

The global picture

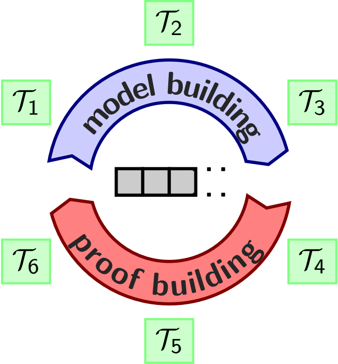
...is roughly the same as before (all theories somehow participate to the main conflict-driven loop):



...except that it it now **parametric** in $\mathcal{T}_1, \ldots, \mathcal{T}_n$.
The trail is made of single assignments $t \leftarrow \mathfrak{c}$ (term+value of matching sorts) coming from different theories (+ some structure).

The global picture

...is roughly the same as before (all theories somehow participate to the main conflict-driven loop):



...except that it it now **parametric** in $\mathcal{T}_1, \ldots, \mathcal{T}_n$.

The trail is made of single assignments $t \leftarrow \mathfrak{c}$ (term+value of matching sorts) coming from different theories (+ some structure).

Everything is on the trail, including assertions from the input problem (e.g. $C \leftarrow$ true for an input clause C)

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

> Typically for MCSAT theories,
> $\mathcal{T}$-values are the domain elements in the canonical model.

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

> Typically for MCSAT theories,
> $\mathcal{T}$-values are the domain elements in the canonical model.

Values can be seen as new constants extending $\mathcal{T}$'s language.

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

> Typically for MCSAT theories,
> $\mathcal{T}$-values are the domain elements in the canonical model.

Values can be seen as new constants extending $\mathcal{T}$'s language.

These new constants satisfy some particular properties w.r.t $\mathcal{T}$ (e.g. $\sqrt{2} \cdot \sqrt{2} \simeq 2$): these are specified in an extension $\mathcal{T}^+$ of $\mathcal{T}$ in the extended language.

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

> Typically for MCSAT theories,
> $\mathcal{T}$-values are the domain elements in the canonical model.

Values can be seen as new constants extending $\mathcal{T}$'s language.

These new constants satisfy some particular properties w.r.t $\mathcal{T}$ (e.g. $\sqrt{2} \cdot \sqrt{2} \simeq 2$): these are specified in an extension $\mathcal{T}^+$ of $\mathcal{T}$ in the extended language. $\mathcal{T}^+$ must be a conservative extension of $\mathcal{T}$ (problems in the original language that are $\mathcal{T}^+$-unsat are $\mathcal{T}$-unsat).

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

Typically for MCSAT theories,
$\mathcal{T}$-values are the domain elements in the canonical model.

Values can be seen as new constants extending $\mathcal{T}$'s language.

These new constants satisfy some particular properties w.r.t $\mathcal{T}$ (e.g. $\sqrt{2} \cdot \sqrt{2} \simeq 2$): these are specified in an extension $\mathcal{T}^+$ of $\mathcal{T}$ in the extended language. $\mathcal{T}^+$ must be a conservative extension of $\mathcal{T}$ (problems in the original language that are $\mathcal{T}^+$-unsat are $\mathcal{T}$-unsat).

We may leave some or all of the sorts without $\mathcal{T}$-values:
$\mathcal{T}$ will not publish on the trail assignments for terms of those sorts.

Where are the values taken from?

For each theory $\mathcal{T}$ to combine, and each sort that it knows of, we must specify a **pool of $\mathcal{T}$-values** to use in assignments: e.g., if we want to solve $(x \cdot x \simeq 2)$, we may want to write $x \leftarrow \sqrt{2}$.

> Typically for MCSAT theories,
> $\mathcal{T}$-values are the domain elements in the canonical model.

Values can be seen as new constants extending $\mathcal{T}$'s language.

These new constants satisfy some particular properties w.r.t $\mathcal{T}$ (e.g. $\sqrt{2} \cdot \sqrt{2} \simeq 2$): these are specified in an extension $\mathcal{T}^+$ of $\mathcal{T}$ in the extended language. $\mathcal{T}^+$ must be a conservative extension of $\mathcal{T}$ (problems in the original language that are $\mathcal{T}^+$-unsat are $\mathcal{T}$-unsat).

We may leave some or all of the sorts without $\mathcal{T}$-values: $\mathcal{T}$ will not publish on the trail assignments for terms of those sorts.

Exception:
every theory uses the two values true and false for sort Bool

What does each theory see of the trail?

When combining $\mathcal{T}$ and $\mathcal{T}'$, if $\mathcal{T}$ writes $u \leftarrow \mathfrak{c}$ on the trail, what can $\mathcal{T}'$ understand from it?

What does each theory see of the trail?

When combining $\mathcal{T}$ and $\mathcal{T}'$, if $\mathcal{T}$ writes $u \leftarrow \mathfrak{c}$ on the trail, what can $\mathcal{T}'$ understand from it?

Not much!

Only that if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}$ and $u_2 \leftarrow \mathfrak{c}$,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \simeq u_2$.

What does each theory see of the trail?

When combining $\mathcal{T}$ and $\mathcal{T}'$, if $\mathcal{T}$ writes $u \leftarrow \mathfrak{c}$ on the trail, what can $\mathcal{T}'$ understand from it?

Not much!

Only that if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}$ and $u_2 \leftarrow \mathfrak{c}$,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \simeq u_2$.

Similarly if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}_1$ and $u_2 \leftarrow \mathfrak{c}_2$ with two distinct values,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \not\simeq u_2$.

What does each theory see of the trail?

When combining $\mathcal{T}$ and $\mathcal{T}'$, if $\mathcal{T}$ writes $u \leftarrow \mathfrak{c}$ on the trail, what can $\mathcal{T}'$ understand from it?

Not much!

Only that if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}$ and $u_2 \leftarrow \mathfrak{c}$,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \simeq u_2$.

Similarly if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}_1$ and $u_2 \leftarrow \mathfrak{c}_2$ with two distinct values,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \not\simeq u_2$.

This is formalised as the **$\mathcal{T}$-view of the trail**
(this is a theoretical concept, no need to eagerly compute the equalities/disequalities at runtime)

What does each theory see of the trail?

When combining $\mathcal{T}$ and $\mathcal{T}'$, if $\mathcal{T}$ writes $u \leftarrow \mathfrak{c}$ on the trail, what can $\mathcal{T}'$ understand from it?

Not much!

Only that if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}$ and $u_2 \leftarrow \mathfrak{c}$,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \simeq u_2$.

Similarly if $\mathcal{T}$ writes $u_1 \leftarrow \mathfrak{c}_1$ and $u_2 \leftarrow \mathfrak{c}_2$ with two distinct values,
$\mathcal{T}'$ understands the trail as if it contained $u_1 \not\simeq u_2$.

This is formalised as the **$\mathcal{T}$-view of the trail**
(this is a theoretical concept, no need to eagerly compute the equalities/disequalities at runtime)

Exception: all theories understand Boolean assignments

What is a theory module?

A set of inferences of the form

$$t_1 \leftarrow \mathfrak{c}_1, \ldots, t_k \leftarrow \mathfrak{c}_k \vdash l \leftarrow \mathfrak{b}$$

where

- each $t_i \leftarrow \mathfrak{c}_i$ is a single $\mathcal{T}$-assignment
  (a term and a $\mathcal{T}$-value of matching sorts)
- $l \leftarrow \mathfrak{b}$ is a single Boolean assignment
  (a term of sort Bool and a truth value)

What is a theory module?

A set of inferences of the form

$$t_1 \leftarrow \mathfrak{c}_1, \ldots, t_k \leftarrow \mathfrak{c}_k \vdash l \leftarrow \mathfrak{b}$$

where

- each $t_i \leftarrow \mathfrak{c}_i$ is a single $\mathcal{T}$-assignment (a term and a $\mathcal{T}$-value of matching sorts)
- $l \leftarrow \mathfrak{b}$ is a single Boolean assignment (a term of sort Bool and a truth value)
- Soundness requirement:
  Every model of the premisses is a model of the conclusion

What is a theory module?

A set of inferences of the form

$$t_1 \leftarrow \mathfrak{c}_1, \ldots, t_k \leftarrow \mathfrak{c}_k \vdash l \leftarrow \mathfrak{b}$$

where

- each $t_i \leftarrow \mathfrak{c}_i$ is a single $\mathcal{T}$-assignment
  (a term and a $\mathcal{T}$-value of matching sorts)
- $l \leftarrow \mathfrak{b}$ is a single Boolean assignment
  (a term of sort Bool and a truth value)
- Soundness requirement:
  Every model of the premisses is a model of the conclusion
  i.e. any $\mathcal{T}^+$-model of $t_1 {\simeq} \mathfrak{c}_1 \wedge \ldots \wedge t_k {\simeq} \mathfrak{c}_k$ is a model of $l {\simeq} \mathfrak{b}$

What is a theory module?

A set of inferences of the form

$$t_1 \leftarrow \mathfrak{c}_1, \ldots, t_k \leftarrow \mathfrak{c}_k \vdash l \leftarrow \mathfrak{b}$$

where

- each $t_i \leftarrow \mathfrak{c}_i$ is a single $\mathcal{T}$-assignment
  (a term and a $\mathcal{T}$-value of matching sorts)
- $l \leftarrow \mathfrak{b}$ is a single Boolean assignment
  (a term of sort Bool and a truth value)
- Soundness requirement:
  Every model of the premisses is a model of the conclusion
  i.e. any $\mathcal{T}^+$-model of $t_1 \simeq \mathfrak{c}_1 \wedge \ldots \wedge t_k \simeq \mathfrak{c}_k$ is a model of $l \simeq \mathfrak{b}$

Example: $(x \leftarrow \sqrt{2}), (y \leftarrow \sqrt{2}) \vdash x \cdot y \simeq 2$ (evaluation inference)

What is a theory module?

A set of inferences of the form

$$t_1 \leftarrow \mathfrak{c}_1, \ldots, t_k \leftarrow \mathfrak{c}_k \vdash l \leftarrow \mathfrak{b}$$

where

- each $t_i \leftarrow \mathfrak{c}_i$ is a single $\mathcal{T}$-assignment
  (a term and a $\mathcal{T}$-value of matching sorts)
- $l \leftarrow \mathfrak{b}$ is a single Boolean assignment
  (a term of sort Bool and a truth value)
- Soundness requirement:
  Every model of the premisses is a model of the conclusion
  i.e. any $\mathcal{T}^+$-model of $t_1 {\simeq} \mathfrak{c}_1 \wedge \ldots \wedge t_k {\simeq} \mathfrak{c}_k$* is a model of $l {\simeq} \mathfrak{b}$

Example: $(x \leftarrow \sqrt{2}), (y \leftarrow \sqrt{2}) \vdash x \cdot y \simeq 2$ (evaluation inference)

*that interprets distinct constants within $\mathfrak{c}_1, \ldots, \mathfrak{c}_k$ by distinct elements

What is a theory module? (Equality inferences)

All theory modules have the equality inferences:

$$
\begin{array}{rcll}
t_1 \leftarrow \mathfrak{c}_1, t_2 \leftarrow \mathfrak{c}_2 & \vdash & t_1 \simeq t_2 & \text{if } \mathfrak{c}_1 \text{ and } \mathfrak{c}_2 \text{ are the same value} \\
t_1 \leftarrow \mathfrak{c}_1, t_2 \leftarrow \mathfrak{c}_2 & \vdash & t_1 \not\simeq t_2 & \text{if } \mathfrak{c}_1 \text{ and } \mathfrak{c}_2 \text{ are distinct values} \\
\\
 & \vdash & t_1 \simeq t_1 & \\
t_1 \simeq t_2 & \vdash & t_2 \simeq t_1 & \\
t_1 \simeq t_2, t_2 \simeq t_3 & \vdash & t_1 \simeq t_3 &
\end{array}
$$

Trail

...is a stack of justified assignments $_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions $_{?}(t{\leftarrow}\mathfrak{c})$
Justification H: a set of assignments that appear earlier on the trail

Trail initialised with input problem
(assignments with empty justifications).

Example (trail grows downwards):

$(l{\leftarrow}\text{true})$ abbreviated as l

id	trail items	just.
0	$-2{\cdot}x - y < 0$	$\{\}$
1	$x + y < 0$	$\{\}$
2	$x < -1$	$\{\}$
3	$y{\leftarrow}0$	?
4	$-y < -2$	$\{0, 2\}$

Trail

…is a stack of justified assignments ${}_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions ${}_{?}(t{\leftarrow}\mathfrak{c})$
Justification H: a set of assignments that appear earlier on the trail

Trail initialised with input problem
(assignments with empty justifications).

Example (trail grows downwards):

($l{\leftarrow}$true) abbreviated as l

Level:
greatest decision involved

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$y{\leftarrow}0$	?	1
4	$-y < -2$	$\{0, 2\}$	0

Trail

…is a stack of justified assignments $_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions $_{?}(t \leftarrow \mathfrak{c})$
Justification H: a set of assignments that appear earlier on the trail

Trail initialised with input problem
(assignments with empty justifications).

Example (trail grows downwards):

$(l \leftarrow \text{true})$ abbreviated as l

Level:
greatest decision involved

Here: conflict of level 1
(if conflict is of level 0…
…problem is unsat)

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$\mathbf{-y < -2}$	$\{0, 2\}$	0

CDSAT: Search rules

Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$$\Gamma \longrightarrow \Gamma, {}_{?}(t{\leftarrow}\mathfrak{c})$$

Deduce

$$\Gamma \longrightarrow \Gamma, {}_{J\vdash}(t{\leftarrow}\mathfrak{b}) \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is not in } \Gamma,$$

Conflict

$$\Gamma \longrightarrow \langle \Gamma; J, (t{\leftarrow}\overline{\mathfrak{b}}) \rangle \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is in } \Gamma$$

CDSAT: Search rules

Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$$\Gamma \longrightarrow \Gamma, {}_{?}(t{\leftarrow}\mathfrak{c})$$

Deduce

$$\Gamma \longrightarrow \Gamma, {}_{J\vdash}(t{\leftarrow}\mathfrak{b}) \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is not in } \Gamma,$$

Conflict

$$\Gamma \longrightarrow \langle \Gamma; J, (t{\leftarrow}\overline{\mathfrak{b}})\rangle \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is in } \Gamma$$

Conflict states $\langle \Gamma; E\rangle$ (E conflicting set of assignments from Γ) are subject to conflict-solving rules similar to MCSAT and CDCL, like resolve:

$$\langle \Gamma; E, (t{\leftarrow}\mathfrak{c})\rangle \longrightarrow \langle \Gamma; E \cup H\rangle \quad \text{if } {}_{H\vdash}(t{\leftarrow}\mathfrak{c}) \text{ is in } \Gamma \text{ and}\ldots$$

CDSAT: Search rules

Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$$\Gamma \longrightarrow \Gamma, {}_{?}(t{\leftarrow}\mathfrak{c})$$

Deduce

$$\Gamma \longrightarrow \Gamma, {}_{J\vdash}(t{\leftarrow}\mathfrak{b}) \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma,$$

and $t{\leftarrow}\overline{\mathfrak{b}}$ is not in Γ,

and t is in $\mathcal{B}$

Conflict

$$\Gamma \longrightarrow \langle \Gamma; J, (t{\leftarrow}\overline{\mathfrak{b}}) \rangle \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma,$$

and $t{\leftarrow}\overline{\mathfrak{b}}$ is in Γ

Conflict states $\langle \Gamma; E \rangle$ (E conflicting set of assignments from Γ) are subject to conflict-solving rules similar to MCSAT and CDCL, like resolve:

$$\langle \Gamma; E, (t{\leftarrow}\mathfrak{c}) \rangle \longrightarrow \langle \Gamma; E \cup H \rangle \quad \text{if } {}_{H\vdash}(t{\leftarrow}\mathfrak{c}) \text{ is in } \Gamma \text{ and} \ldots$$

CDSAT: Search rules

CDSAT is parameterized by finite set of terms $\mathcal{B}$ called global basis.
Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$$\Gamma \longrightarrow \Gamma, {}_{?}(t \leftarrow \mathfrak{c})$$

Deduce

$$\Gamma \longrightarrow \Gamma, {}_{J\vdash}(t \leftarrow \mathfrak{b}) \quad \text{if } J \vdash_{\mathcal{T}} (t \leftarrow \mathfrak{b}) \text{ and } J \subseteq \Gamma,$$

and $t \leftarrow \overline{\mathfrak{b}}$ is not in Γ,
and t is in $\mathcal{B}$

Conflict

$$\Gamma \longrightarrow \langle \Gamma; J, (t \leftarrow \overline{\mathfrak{b}}) \rangle \quad \text{if } J \vdash_{\mathcal{T}} (t \leftarrow \mathfrak{b}) \text{ and } J \subseteq \Gamma,$$

and $t \leftarrow \overline{\mathfrak{b}}$ is in Γ

Conflict states $\langle \Gamma; E \rangle$ (E conflicting set of assignments from Γ) are subject to conflict-solving rules similar to MCSAT and CDCL, like resolve:

$$\langle \Gamma; E, (t \leftarrow \mathfrak{c}) \rangle \longrightarrow \langle \Gamma; E \cup H \rangle \quad \text{if } {}_{H\vdash}(t \leftarrow \mathfrak{c}) \text{ is in } \Gamma \text{ and} \ldots$$

CDSAT: Search rules

CDSAT is parameterized by finite set of terms $\mathcal{B}$ called global basis.
Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$$\Gamma \longrightarrow \Gamma, {}_{?}(t{\leftarrow}\mathfrak{c}) \qquad \text{if } t{\leftarrow}\mathfrak{c} \text{ is “relevant \& acceptable” given } \mathcal{T}\text{'s view of the trail } \Gamma$$

Deduce

$$\Gamma \longrightarrow \Gamma, {}_{J\vdash}(t{\leftarrow}\mathfrak{b}) \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is not in } \Gamma, \text{ and } t \text{ is in } \mathcal{B}$$

Conflict

$$\Gamma \longrightarrow \langle \Gamma; J, (t{\leftarrow}\overline{\mathfrak{b}}) \rangle \qquad \text{if } J \vdash_{\mathcal{T}} (t{\leftarrow}\mathfrak{b}) \text{ and } J \subseteq \Gamma, \text{ and } t{\leftarrow}\overline{\mathfrak{b}} \text{ is in } \Gamma$$

Conflict states $\langle \Gamma; E \rangle$ (E conflicting set of assignments from Γ) are subject to conflict-solving rules similar to MCSAT and CDCL, like resolve:

$$\langle \Gamma; E, (t{\leftarrow}\mathfrak{c}) \rangle \longrightarrow \langle \Gamma; E \cup H \rangle \quad \text{if } {}_{H\vdash}(t{\leftarrow}\mathfrak{c}) \text{ is in } \Gamma \text{ and} \ldots$$

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w \ , \ w - 2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \ , \ w - 2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w \ , \ w - 2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u {\leftarrow} \mathfrak{c}$	?	1
5	$v {\leftarrow} \mathfrak{c}$	?	2

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\, v][j]) \simeq w \;,\; w-2 \simeq f(u) \;,\; i \simeq j \;,\; u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\, v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\, v][j] \leftarrow \mathfrak{c}$	?	3

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u {\leftarrow} \mathfrak{c}$	?	1
5	$v {\leftarrow} \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] {\leftarrow} \mathfrak{c}$	?	3
7	$w {\leftarrow} 0$	?	4

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\,v][j]) \leftarrow 0$	?	5

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \ , \ w - 2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\,v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\,v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$u \simeq a[i{:=}\,v][j]$	$\{4,6\}$	3

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$w{\leftarrow}0$	?	4
8	$f(a[i{:=}\ v][j]){\leftarrow}0$	?	5
9	$f(u){\leftarrow}{-2}$	?	6
10	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
11	$f(u) \not\simeq f(a[i{:=}\ v][j])$	$\{8,9\}$	6

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ \ w-2 \simeq f(u)\ ,\ \ i \simeq j\ ,\ \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\ v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$\mathbf{u} \simeq \mathbf{a}[\mathbf{i}{:=}\ \mathbf{v}][\mathbf{j}]$	$\{4, 6\}$	3
11	$\mathbf{f}(\mathbf{u}) \not\simeq \mathbf{f}(\mathbf{a}[\mathbf{i}{:=}\ \mathbf{v}][\mathbf{j}])$	$\{8, 9\}$	6
	conflict E^1: $\{10, 11\}$		6

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w \ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\ v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$\mathbf{u \simeq a[i{:=}\ v][j]}$	$\{4,6\}$	3
11	$\mathbf{f(u) \not\simeq f(a[i{:=}\ v][j])}$	$\{8,9\}$	6
	conflict E^1: $\{10,11\}$		6

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ \ w - 2 \simeq f(u)\ ,\ \ i \simeq j\ ,\ \ u \simeq v$$

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$w{\leftarrow}0$	?	4
8	$f(a[i{:=}\ v][j]){\leftarrow}0$	?	5
9	$f(u){\leftarrow}{-2}$	?	6
10	$\mathbf{u \simeq a[i{:=}\ v][j]}$	$\{4,6\}$	3
11	$\mathbf{f(u) \not\simeq f(a[i{:=}\ v][j])}$	$\{8,9\}$	6
	conflict E^1: $\{10,11\}$		6

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3

...

3. Termination, Soundness and Completeness

Termination and Soundness

Termination:

Theorem: If the global basis $\mathcal{B}$ is finite, CDSAT terminates.

Termination and Soundness

Termination:

Theorem: If the global basis $\mathcal{B}$ is finite, CDSAT terminates.

How to determine $\mathcal{B}$? It should be sufficiently large to allow each theory module to explain its conflicts via deductions.

Termination and Soundness

Termination:

Theorem: If the global basis $\mathcal{B}$ is finite, CDSAT terminates.

How to determine $\mathcal{B}$? It should be sufficiently large to allow each theory module to explain its conflicts via deductions.

For each theory module $\mathcal{T}$ involved,
and all finite sets X of terms (think of it as the terms of the input),
we must have a finite set of terms $\mathrm{basis}_{\mathcal{T}}(X)$, called local basis
(those terms possibly introduced by $\mathcal{T}$ during the run)

Termination and Soundness

Termination:

Theorem: If the global basis $\mathcal{B}$ is finite, CDSAT terminates.

How to determine $\mathcal{B}$? It should be sufficiently large to allow each theory module to explain its conflicts via deductions.

For each theory module $\mathcal{T}$ involved,
and all finite sets X of terms (think of it as the terms of the input),
we must have a finite set of terms $\text{basis}_{\mathcal{T}}(X)$, called local basis
(those terms possibly introduced by $\mathcal{T}$ during the run)

If the local bases of $\mathcal{T}_1, \ldots, \mathcal{T}_n$ satisfy some (collective) properties,
then it is possible to define a finite global basis $\mathcal{B}$ for $\bigcup_{k=1}^{n} \mathcal{T}_k$.

Termination and Soundness

Termination:

Theorem: If the global basis $\mathcal{B}$ is finite, CDSAT terminates.

How to determine $\mathcal{B}$? It should be sufficiently large to allow each theory module to explain its conflicts via deductions.

For each theory module $\mathcal{T}$ involved,
and all finite sets X of terms (think of it as the terms of the input),
we must have a finite set of terms $\text{basis}_{\mathcal{T}}(X)$, called local basis
(those terms possibly introduced by $\mathcal{T}$ during the run)

If the local bases of $\mathcal{T}_1, \ldots, \mathcal{T}_n$ satisfy some (collective) properties,
then it is possible to define a finite global basis $\mathcal{B}$ for $\bigcup_{k=1}^{n} \mathcal{T}_k$.

Soundness:

Theorem: Since each theory module $\mathcal{T}$ is made of sound inferences,
if the calculus ends with a conflict of level 0,
then the input was unsat.
(you can even get a proof)

What happens if we never get unsat?

Do we have a model?

What happens if we never get unsat?

Do we have a model?

This relies on a completeness condition for theory modules:

A $\mathcal{T}$-module is complete if for any Γ,

- Either There exists a $\mathcal{T}^+$-model of the theory view of Γ
- Or $\mathcal{T}$ can make a (relevant & acceptable) decision
- Or a $\mathcal{T}$-inference can deduce a new assignment (for a term in the local basis)

What happens if we never get unsat?

Do we have a model?

This relies on a completeness condition for theory modules:

A $\mathcal{T}$-module is complete if for any Γ,

- Either There exists a $\mathcal{T}^+$-model of the theory view of Γ
- Or $\mathcal{T}$ can make a (relevant & acceptable) decision
- Or a $\mathcal{T}$-inference can deduce a new assignment (for a term in the local basis)

In a combination though, the $\mathcal{T}_k$-models have to agree on the sorts' cardinalities and equalities between shared variables/terms.

What happens if we never get unsat?

Do we have a model?

This relies on a completeness condition for theory modules:

A $\mathcal{T}$-module is complete if for any Γ,

- Either There exists a $\mathcal{T}^+$-model of the theory view of Γ
- Or $\mathcal{T}$ can make a (relevant & acceptable) decision
- Or a $\mathcal{T}$-inference can deduce a new assignment (for a term in the local basis)

In a combination though, the $\mathcal{T}_k$-models have to agree on the sorts' cardinalities and equalities between shared variables/terms.

The paper has a version of completeness that takes care of this: $\mathcal{T}_0$-completeness, where $\mathcal{T}_0$ is a reference theory that can be used to synchronise cardinalities (for a combination of stably infinite theories, take $\mathcal{T}_0$ to force the interpretation of all sorts to be $\mathbb{N}$).

What happens if we never get unsat?

Do we have a model?

This relies on a completeness condition for theory modules:

A $\mathcal{T}$-module is complete if for any Γ,

- Either There exists a $\mathcal{T}^+$-model of the theory view of Γ
- Or $\mathcal{T}$ can make a (relevant & acceptable) decision
- Or a $\mathcal{T}$-inference can deduce a new assignment (for a term in the local basis)

In a combination though, the $\mathcal{T}_k$-models have to agree on the sorts' cardinalities and equalities between shared variables/terms.

The paper has a version of completeness that takes care of this: $\mathcal{T}_0$-completeness, where $\mathcal{T}_0$ is a reference theory that can be used to synchronise cardinalities (for a combination of stably infinite theories, take $\mathcal{T}_0$ to force the interpretation of all sorts to be $\mathbb{N}$).

Theorem: Assume $\mathcal{T}_0$ has a complete module, and all other theories have $\mathcal{T}_0$-complete modules.

If CDSAT cannot make any further transitions, then the trail describes a model for the union of the (extended) theories.

Theory modules given as examples in the paper

- EUF

$$(t_i \simeq u_i)_{i=1\ldots n}, (f(t_1, \ldots, t_n) \not\simeq f(u_1, \ldots, u_n)) \vdash_{\mathsf{EUF}} \quad \bot$$

- Arrays: similar, except for extensionality

- LRA: evaluation inference, Fourier-Motzkin resolution inference as in MCSAT, etc

Theory modules given as examples in the paper

- EUF

$$(t_i \simeq u_i)_{i=1\ldots n}, (f(t_1, \ldots, t_n) \not\simeq f(u_1, \ldots, u_n)) \vdash_{\mathsf{EUF}} \quad \bot$$

- Arrays: similar, except for extensionality

- LRA: evaluation inference, Fourier-Motzkin resolution inference as in MCSAT, etc

- Black box procedure for equality-sharing: coarse-grain inferences

$$l_1 \leftarrow \mathfrak{b}_1, \ldots, l_n \leftarrow \mathfrak{b}_n \vdash_{\mathcal{T}} \bot$$

  where $l_1, \ldots, l_n$ are formulæ, and the conjunction of the literals corresponding to the Boolean assignments $l_1 \leftarrow b_1, \ldots, l_n \leftarrow b_n$ is $\mathcal{T}$-unsatisfiable (as detected by the black box)

Theory modules given as examples in the paper

- EUF ($\mathcal{T}_0$-complete for all $\mathcal{T}_0$)

$$(t_i \simeq u_i)_{i=1\ldots n}, (f(t_1,\ldots,t_n) \not\simeq f(u_1,\ldots,u_n)) \vdash_{\mathsf{EUF}} \quad \bot$$

- Arrays: similar, except for extensionality
  ($\mathcal{T}_0$-complete for all $\mathcal{T}_0$ such that...)
- LRA: evaluation inference, Fourier-Motzkin resolution inference as in MCSAT, etc
  ($\mathcal{T}_0$-complete for all $\mathcal{T}_0$ imposing $|Q|$ infinite)
- Black box procedure for equality-sharing: coarse-grain inferences

$$l_1 \leftarrow \mathfrak{b}_1, \ldots, l_n \leftarrow \mathfrak{b}_n \vdash_{\mathcal{T}} \bot$$

  where $l_1, \ldots, l_n$ are formulæ, and the conjunction of the literals corresponding to the Boolean assignments $l_1 \leftarrow b_1, \ldots, l_n \leftarrow b_n$ is $\mathcal{T}$-unsatisfiable (as detected by the black box)
  ($\mathcal{T}_0$-complete for all $\mathcal{T}_0$ imposing the cardinality of all known sorts but Bool to be countably infinite)

Concluding remarks

- Learning:
  Not needed for soundness, completeness, and termination, but highly desirable - in the paper's long version
- Proof production: is easy, each theory inference can come with a proof object, CDSAT only aggregates them in simple ways
- CDSAT is a framework:
  leaves large freedom to the design of search plans / strategies
- First-order assignments: I mostly presented them as a way to build a model of an input formula - they could be part of the input

$$l_1 \leftarrow \mathfrak{b}_1, \ldots, l_k \leftarrow \mathfrak{b}_k, t_1 \leftarrow \mathfrak{c}_1, \ldots, t_j \leftarrow \mathfrak{c}_j$$

  The question is then "Is there a model of the constraints (in sort Bool) that extends these first-order assignments?"
  Note: the choice of theory extensions impacts the meaning of the question.
  We suggest to call this SMA,
  for Satisfiability Modulo Assignments.

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool
  Now needs to be populated by other theories

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool
  Now needs to be populated by other theories

- Lay down on paper:
  how a single E-graph can factor equality reasoning in CDSAT.
  The trail + E-graph become the front-end of architecture
  (as opposed to DPLL($\mathcal{T}$) where it is the SAT-solver)

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool
  Now needs to be populated by other theories

- Lay down on paper:
  how a single E-graph can factor equality reasoning in CDSAT.
  The trail + E-graph become the front-end of architecture
  (as opposed to DPLL($\mathcal{T}$) where it is the SAT-solver)

- Non-disjoint theories?

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool
  Now needs to be populated by other theories

- Lay down on paper:
  how a single E-graph can factor equality reasoning in CDSAT.
  The trail + E-graph become the front-end of architecture
  (as opposed to DPLL($\mathcal{T}$) where it is the SAT-solver)

- Non-disjoint theories?

- How to handle quantifiers?
  Technically, MCSAT has to do with quantifier elimination.
  How can this be exploited for quantified problems in combinations of theories?

Further work

- State of the implementation:
  An OCaml prototype implements the CDSAT framework (with learning), with theory module Bool
  Now needs to be populated by other theories

- Lay down on paper:
  how a single E-graph can factor equality reasoning in CDSAT.
  The trail + E-graph become the front-end of architecture
  (as opposed to DPLL($\mathcal{T}$) where it is the SAT-solver)

- Non-disjoint theories?

- How to handle quantifiers?
  Technically, MCSAT has to do with quantifier elimination.
  How can this be exploited for quantified problems in combinations of theories?

C. Barrett, R. Nieuwenhuis, A. Oliveras, and C. Tinelli.
Splitting on demand in SAT Modulo Theories.
In M. Hermann and A. Voronkov, editors, *Proc. of the the 13th Int. Conf. on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR'06)*, volume 4246 of *LNCS*, pages 512–526. Springer-Verlag, 2006.

S. Cotton.
Natural domain SMT: A preliminary assessment.
In K. Chatterjee and T. A. Henzinger, editors, *Proceedings of the Eighth International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS)*, volume 6246 of *Lecture Notes in Computer Science*, pages 77–91. Springer, 2010.

L. M. de Moura and D. Jovanovic.
A model-constructing satisfiability calculus.
In R. Giacobazzi, J. Berdine, and I. Mastroeni, editors, *Proc. of the 14th Int. Conf. on Verification, Model Checking, and Abstract Interpretation (VMCAI'13)*, volume 7737 of *LNCS*, pages 1–12. Springer-Verlag, 2013.

S. Graham-Lengrand and D. Jovanović.
An MCSAT treatment of bit-vectors.
In M. Brain and L. Hadarean, editors, *15 Int. Work. on Satisfiability Modulo Theories (SMT 2017)*, 2017.

D. Jovanović, C. Barrett, and L. de Moura.
The design and implementation of the model constructing satisfiability calculus.
In *Proc. of the 13th Int. Conf. on Formal Methods In Computer-Aided Design (FMCAD '13)*. FMCAD Inc., 2013. Portland, Oregon

D. Jovanović and L. de Moura.
Cutting to the chase: solving linear integer arithmetic.
In N. Bjørner and V. Sofronie-Stokkermans, editors, *Proc. of the 23rd Int. Conf. on Automated Deduction (CADE'11)*, volume 6803 of *LNCS*, pages 338–353. Springer-Verlag, 2011.

D. Jovanović and L. de Moura.
Solving non-linear arithmetic.
In B. Gramlich, D. Miller, and U. Sattler, editors, *Proc. of the 6th Int. Joint Conf. on Automated Reasoning (IJCAR'12)*, volume 7364 of *LNCS*, pages 339–354. Springer-Verlag, 2012.

D. Jovanović.
Solving nonlinear integer arithmetic with MCSAT.
In A. Bouajjani and D. Monniaux, editors, *Proc. of the 18th Int. Conf. on Verification, Model Checking, and Abstract Interpretation (VMCAI'17)*, volume 10145 of *LNCS*, pages 330–346. Springer-Verlag, 2017.

K. Korovin, N. Tsiskaridze, and A. Voronkov.
Conflict resolution.
In I. P. Gent, editor, *Proceedings of the Fifteenth International Conference on Principles and Practice of Constraint Programming (CP)*, volume 5732 of *Lecture Notes in Computer Science*, pages 509–523. Springer, 2009.

K. L. McMillan, A. Kuehlmann, and M. Sagiv.
Generalizing DPLL to richer logics.
In A. Bouajjani and O. Maler, editors, *Proceedings of the Twenty-First International Conference on Computer Aided Verification (CAV)*, volume 5643 of *Lecture Notes in Computer Science*, pages 462–476. Springer, 2009.

G. Nelson and D. C. Oppen.
Simplification by cooperating decision procedures.
ACM Press Trans. on Program. Lang. and Syst., 1(2):245–257, 1979.

C. Wang, F. Ivančić, M. Ganai, and A. Gupta.
Deciding separation logic formulae by SAT and incremental negative cycle elimination.
In G. Sutcliffe and A. Voronkov, editors, *Proceedings of the Twelfth International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, volume 3835 of *Lecture Notes in Artificial Intelligence*, pages 322–336. Springer, 2005.

S. A. Wolfman and D. S. Weld.
The LPSAT engine and its application to resource planning.
In T. Dean, editor, *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, volume 1, pages 310–316. Morgan Kaufmann Publishers, 1999.

A. Zeljic, C. M. Wintersteiger, and P. Rümmer.
Deciding bit-vector formulas with mcsat.
In N. Creignou and D. L. Berre, editors, *Proc. of the 19th Int. Conf. on Theory and Applications of Satisfiability Testing (RTA'06)*, volume 9710 of *LNCS*, pages 249–266. Springer-Verlag, 2016.

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0), \qquad l_1 : (x + y < 0), \qquad l_2 : (x < -1)$

unsatisfiable in LRA.

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$?

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$

An example in Linear Rational Arithmetic

$l_0 : (-2\cdot x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.

An example in Linear Rational Arithmetic

$l_0 : (-2 \cdot x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (- y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to infer $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y \leftarrow 4$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $\quad l_1 : (x+y < 0)$, $\quad l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (- y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to infer $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y \leftarrow 4$
- Undo guess, keep l_4

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to infer $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y \leftarrow 4$
- Undo guess, keep l_4
  l_3 and l_4 give clash of bounds for y

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x - y < 0)$, $l_1 : (x + y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y \leftarrow 0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y \leftarrow 0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y \leftarrow 0$
- Now undo the guess but keep l_3.
- Try new guess, say $y \leftarrow 4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to infer $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y \leftarrow 4$
- Undo guess, keep l_4
  l_3 and l_4 give clash of bounds for y
- Suggests to infer $l_3 + l_4$, i.e. $l_5 : 0 < -2$

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x{-}y < 0)$, $l_1 : (x{+}y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- Guess a value, e.g. $y{\leftarrow}0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y{\leftarrow}0$? No:
- Clash of bounds suggests to infer $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y{\leftarrow}0$
- Now undo the guess but keep l_3.
- Try new guess, say $y{\leftarrow}4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to infer $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y{\leftarrow}4$
- Undo guess, keep l_4
  l_3 and l_4 give clash of bounds for y
- Suggests to infer $l_3 + l_4$, i.e. $l_5 : 0 < -2$
  No guess to undo, problem is UNSAT

An example in Linear Rational Arithmetic

$l_0 : (-2{\cdot}x-y < 0)$, $l_1 : (x+y < 0)$, $l_2 : (x < -1)$

unsatisfiable in LRA.

- **Guess a value**, e.g. $y{\leftarrow}0$
  Then l_0 yields lower bound $x > 0$
  Together with l_2, space of possible values for x is empty
  What to do? just undo $y{\leftarrow}0$? No:
- Clash of bounds suggests to **infer** $l_0 + 2l_2$, i.e. $l_3 : (-y < -2)$
  indeed violated by the guess $y{\leftarrow}0$
- Now **undo the guess** but keep l_3.
- Try new **guess**, say $y{\leftarrow}4$
  l_1 yields upper bound $x < -4$, l_0 yields lower bound $x > -2$
- Clash of bounds suggests to **infer** $l_0 + 2l_1$, i.e. $l_4 : (y < 0)$
  indeed violated by the guess $y{\leftarrow}4$
- **Undo guess**, keep l_4
  l_3 and l_4 give clash of bounds for y
- Suggests to **infer** $l_3 + l_4$, i.e. $l_5 : 0 < -2$
  No guess to undo, problem is UNSAT

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions ${}_{?}(t{\leftarrow}\mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l{\leftarrow}\text{true})$ abbrev. as l

id	trail items	just.
0	$-2{\cdot}x - y < 0$	$\{\}$
1	$x + y < 0$	$\{\}$
2	$x < -1$	$\{\}$

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t\leftarrow\mathfrak{c})$ and decisions ${}_{?}(t\leftarrow\mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l\leftarrow\text{true})$ abbrev. as l

Level:
greatest decision involved

id	trail items	just.	lev.
0	$-2\cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$y\leftarrow 0$	?	1

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions ${}_{?}(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$y \leftarrow 0$	?	1
4	$-y < -2$	$\{0, 2\}$	0

Trail

Trail = stack of justified assignments ${}_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions ${}_{?}(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$-\mathbf{y} < \mathbf{-2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Trail

Trail = stack of justified assignments ${}_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions ${}_{?}(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$-\mathbf{y} < -\mathbf{2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0

Trail

Trail $=$ stack of justified assignments $_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions $_{?}(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2 \cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$\mathbf{- y < -2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2 \cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$y \leftarrow 4$	?	1

Trail

Trail $=$ stack of justified assignments $_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions $_{?}(t{\leftarrow}\mathfrak{c})$, Trail initialised with input problem (assign. with empty justifications)

$(l{\leftarrow}\text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y{\leftarrow}0}$	?	1
4	$\mathbf{-y < -2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$y{\leftarrow}4$	?	1
5	$y < 0$	$\{0, 1\}$	0

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions ${}_{?}(t{\leftarrow}\mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l{\leftarrow}\text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y{\leftarrow}0}$	?	1
4	$-\mathbf{y} < \mathbf{-2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$\mathbf{y{\leftarrow}4}$	?	1
5	$\mathbf{y} < \mathbf{0}$	$\{0, 1\}$	0
	conflict E^2: $\{4, 5\}$		1

Trail

Trail $=$ stack of justified assignments $_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions $_?(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0…
…problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$\mathbf{-y < -2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$\mathbf{y \leftarrow 4}$	?	1
5	$\mathbf{y < 0}$	$\{0, 1\}$	0
	conflict E^2: $\{4, 5\}$		1

Phase 3

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$y < 0$	$\{0, 1\}$	0

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t{\leftarrow}\mathfrak{c})$ and decisions ${}_{?}(t{\leftarrow}\mathfrak{c})$, Trail initialised with input problem (assign. with empty justifications)

$(l{\leftarrow}\text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0...
...problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y{\leftarrow}0}$	?	1
4	$\mathbf{-y < -2}$	$\{0,2\}$	0
	conflict E^1: $\{3,4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0,2\}$	0
4	$\mathbf{y{\leftarrow}4}$	?	1
5	$\mathbf{y < 0}$	$\{0,1\}$	0
	conflict E^2: $\{4,5\}$		1

Phase 3

id	trail items	just.	lev.
0	$-2{\cdot}x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0,2\}$	0
4	$y < 0$	$\{0,1\}$	0
5	$0 < -2$	$\{3,4\}$	0

Trail

Trail $=$ stack of justified assignments ${}_{H\vdash}(t \leftarrow \mathfrak{c})$ and decisions ${}_{?}(t \leftarrow \mathfrak{c})$,
Trail initialised with input problem (assign. with empty justifications)

$(l \leftarrow \text{true})$ abbrev. as l

Level:
greatest decision involved

If conflict is of level 0…
…problem is unsat

Phase 1

id	trail items	just.	lev.
0	$-2 \cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$\mathbf{y \leftarrow 0}$	?	1
4	$\mathbf{-y < -2}$	$\{0, 2\}$	0
	conflict E^1: $\{3, 4\}$		1

Phase 2

id	trail items	just.	lev.
0	$-2 \cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$\mathbf{y \leftarrow 4}$	?	1
5	$\mathbf{y < 0}$	$\{0, 1\}$	0
	conflict E^2: $\{4, 5\}$		1

Phase 3

id	trail items	just.	lev.
0	$-2 \cdot x - y < 0$	$\{\}$	0
1	$x + y < 0$	$\{\}$	0
2	$x < -1$	$\{\}$	0
3	$-y < -2$	$\{0, 2\}$	0
4	$y < 0$	$\{0, 1\}$	0
5	$\mathbf{0 < -2}$	$\{3, 4\}$	0
	conflict E^3: $\{5\}$		0

CDSAT: Search rules

Parameterized by finite set of terms $\mathcal{B}$ called global basis
Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$\Gamma \longrightarrow \Gamma, {}_{?}(t \leftarrow \mathfrak{c})$ if $t \leftarrow \mathfrak{c}$ (in $\mathcal{T}$-public sort) does not immediately violate $\mathcal{T}$'s view of the trail $\Gamma_{\mathcal{T}}$

Deduce

$\Gamma \longrightarrow \Gamma, {}_{J\vdash} L$ if $J \vdash_{\mathcal{T}} L$ and $J \subseteq \Gamma$, and $\overline{L}$ is not in Γ, and L is for a formula in $\mathcal{B}$

Conflict

$\Gamma \longrightarrow \langle \Gamma; J, \overline{L} \rangle$ if $J \vdash_{\mathcal{T}} L$ and $J \subseteq \Gamma$, and $\overline{L}$ is in Γ

CDSAT: Search rules

Parameterized by finite set of terms $\mathcal{B}$ called global basis
Let $\mathcal{T}$ be a theory with a specific $\mathcal{T}$-module.

Decide

$\Gamma \longrightarrow \Gamma, {}_{?}(t \leftarrow \mathfrak{c})$ if $t \leftarrow \mathfrak{c}$ (in $\mathcal{T}$-public sort) does not immediately violate $\mathcal{T}$'s view of the trail $\Gamma_{\mathcal{T}}$

Deduce

$\Gamma \longrightarrow \Gamma, {}_{J\vdash} L$ if $J \vdash_{\mathcal{T}} L$ and $J \subseteq \Gamma$, and $\overline{L}$ is not in Γ, and L is for a formula in $\mathcal{B}$

Conflict

$\Gamma \longrightarrow \langle \Gamma; J, \overline{L} \rangle$ if $J \vdash_{\mathcal{T}} L$ and $J \subseteq \Gamma$, and $\overline{L}$ is in Γ

CDSAT: Conflict analysis rules

Fail

$\langle \Gamma; \emptyset \rangle \longrightarrow$ unsat

Undo

$\langle \Gamma; E, A \rangle \longrightarrow \Gamma^{\leq m-1}$ if A is a non-Boolean decision of level $m > \text{level}_\Gamma(E)$

Backjump

$\langle \Gamma; E, L \rangle \longrightarrow \Gamma^{\leq m}, {}_{E\vdash}\overline{L}$ if $\text{level}_\Gamma(L) > m$, where $m = \text{level}_\Gamma(E)$

Resolve

$\langle \Gamma; E, A \rangle \longrightarrow \langle \Gamma; E \cup H \rangle$ if ${}_{H\vdash}A$ is in Γ and H does not contain a non-Boolean decision whose level is $\text{level}_\Gamma(E, A)$

UndoDecide

$\langle \Gamma; E, L, L' \rangle \longrightarrow \Gamma^{\leq m-1}, {}_{?}\overline{L}$ if ${}_{H\vdash}L$ and ${}_{H'\vdash}L'$ are in Γ and $H \cap H'$ contains a non-Boolean decision of level $m = \text{level}_\Gamma(E, L, L')$

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \;,\; w-2 \simeq f(u) \;,\; i \simeq j \;,\; u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w - 2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \;,\; w-2 \simeq f(u) \;,\; i \simeq j \;,\; u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \ , \ w - 2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\,v][j]) \leftarrow 0$	?	5

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

	Phase 1		
id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\ v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\ v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\ v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w\ ,\ w-2 \simeq f(u)\ ,\ i \simeq j\ ,\ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\,v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$w{\leftarrow}0$	?	4
8	$f(a[i{:=}\,v][j]){\leftarrow}0$	?	5
9	$f(u){\leftarrow}{-}2$	?	6
10	$u \simeq a[i{:=}\,v][j]$	$\{4,6\}$	3
11	$f(u) \not\simeq f(a[i{:=}\,v][j])$	$\{8,9\}$	6

An example with arithmetic, arrays, congruence

$$f(a[i:=v][j]) \simeq w \ , \ w-2 \simeq f(u) \ , \ i \simeq j \ , \ u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i:=v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i:=v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i:=v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$\mathbf{u} \simeq \mathbf{a}[\mathbf{i}:=\mathbf{v}][\mathbf{j}]$	$\{4, 6\}$	3
11	$\mathbf{f}(\mathbf{u}) \not\simeq \mathbf{f}(\mathbf{a}[\mathbf{i}:=\mathbf{v}][\mathbf{j}])$	$\{8, 9\}$	6
	conflict E^1: $\{10, 11\}$		6

An example with arithmetic, arrays, congruence

$$f(a[i{:=}\,v][j]) \simeq w \;,\; w-2 \simeq f(u) \;,\; i \simeq j \;,\; u \simeq v$$

Phase 1

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$w \leftarrow 0$	?	4
8	$f(a[i{:=}\,v][j]) \leftarrow 0$	?	5
9	$f(u) \leftarrow -2$	?	6
10	$\mathbf{u} \simeq \mathbf{a}[\mathbf{i}{:=}\,\mathbf{v}][\mathbf{j}]$	$\{4,6\}$	3
11	$\mathbf{f}(\mathbf{u}) \not\simeq \mathbf{f}(\mathbf{a}[\mathbf{i}{:=}\,\mathbf{v}][\mathbf{j}])$	$\{8,9\}$	6
	conflict E^1: $\{10,11\}$		6

Phase 2

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\,v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\,v][j])$	$\{7\}$	3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$f(a[i:=v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i:=v][j] \leftarrow \mathfrak{c}$	?	3
7	$u \simeq a[i:=v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i:=v][j])$	$\{7\}$	3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3
10	$w{-}2 \simeq w$	$\{1,9\}$	3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u{\leftarrow}\mathfrak{c}$	?	1
5	$v{\leftarrow}\mathfrak{c}$	?	2
6	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3
10	$\mathbf{w{-}2 \simeq w}$	$\{1,9\}$	3
	conflict E_1^2: $\{10\}$		3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$\mathbf{w{-}2 \simeq f(u)}$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u {\leftarrow} \mathfrak{c}$	?	1
5	$v {\leftarrow} \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] {\leftarrow} \mathfrak{c}$	?	3
7	$u \simeq a[i{:=}\ v][j]$	$\{4, 6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3
9	$\mathbf{f(u) \simeq w}$	$\{0, 8\}$	3
10	$w{-}2 \simeq w$	$\{1, 9\}$	3
	conflict E_1^2: $\{10\}$		3
	conflict E_2^2: $\{1, 9\}$		3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$\mathbf{f(a[i{:}= v][j]) \simeq w}$	$\{\}$	0
1	$\mathbf{w-2 \simeq f(u)}$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:}= v][j] \leftarrow \mathfrak{c}$	?	3
7	$u \simeq a[i{:}= v][j]$	$\{4,6\}$	3
8	$\mathbf{f(u) \simeq f(a[i{:}= v][j])}$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3
10	$w-2 \simeq w$	$\{1,9\}$	3
	conflict E_1^2: $\{10\}$		3
	conflict E_2^2: $\{1,9\}$		3
	conflict E_3^2: $\{0,1,8\}$		3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$\mathbf{f(a[i{:=}\,v][j]) \simeq w}$	$\{\}$	0
1	$\mathbf{w-2 \simeq f(u)}$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\,v][j] \leftarrow \mathfrak{c}$	?	3
7	$\mathbf{u \simeq a[i{:=}\,v][j]}$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\,v][j])$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3
10	$w-2 \simeq w$	$\{1,9\}$	3
	conflict E_1^2: $\{10\}$		3
	conflict E_2^2: $\{1,9\}$		3
	conflict E_3^2: $\{0,1,8\}$		3
	conflict E_4^2: $\{0,1,7\}$		3

An example with arithmetic, arrays, congruence

Phase 2

id	trail items	just.	lev.
0	$\mathbf{f(a[i{:=}\ v][j]) \simeq w}$	$\{\}$	0
1	$\mathbf{w-2 \simeq f(u)}$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \leftarrow \mathfrak{c}$	?	1
5	$v \leftarrow \mathfrak{c}$	?	2
6	$a[i{:=}\ v][j] \leftarrow \mathfrak{c}$	?	3
7	$\mathbf{u \simeq a[i{:=}\ v][j]}$	$\{4,6\}$	3
8	$f(u) \simeq f(a[i{:=}\ v][j])$	$\{7\}$	3
9	$f(u) \simeq w$	$\{0,8\}$	3
10	$w-2 \simeq w$	$\{1,9\}$	3
	conflict E_1^2: $\{10\}$		3
	conflict E_2^2: $\{1,9\}$		3
	conflict E_3^2: $\{0,1,8\}$		3
	conflict E_4^2: $\{0,1,7\}$		3

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0,1\}$	0

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0,1\}$	0

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w - 2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0, 1\}$	0
5	$u \leftarrow \mathfrak{c}$	?	1

An example with arithmetic, arrays, congruence

	Phase 3		
id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0, 1\}$	0
5	$u{\leftarrow}\mathfrak{c}$	?	1
6	$v{\leftarrow}\mathfrak{c}$	?	2

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w-2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0,1\}$	0
5	$u \leftarrow \mathfrak{c}$	?	1
6	$v \leftarrow \mathfrak{c}$	?	2
7	$a[i{:=}\ v][j] \leftarrow \mathfrak{d}$	?	3

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0,1\}$	0
5	$u{\leftarrow}\mathfrak{c}$	?	1
6	$v{\leftarrow}\mathfrak{c}$	?	2
7	$a[i{:=}\ v][j]{\leftarrow}\mathfrak{d}$	?	3
8	$v \not\simeq a[i{:=}\ v][j]$	$\{6,7\}$	3

An example with arithmetic, arrays, congruence

	Phase 3		
id	trail items	just.	lev.
0	$f(a[i{:=}\ v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$\mathbf{i} \simeq \mathbf{j}$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\ v][j]$	$\{0, 1\}$	0
5	$u {\leftarrow} \mathfrak{c}$	?	1
6	$v {\leftarrow} \mathfrak{c}$	?	2
7	$a[i{:=}\ v][j] {\leftarrow} \mathfrak{d}$	?	3
8	$\mathbf{v} \not\simeq \mathbf{a}[\mathbf{i}{:=}\ \mathbf{v}][\mathbf{j}]$	$\{6, 7\}$	3
	conflict E^3: $\{2, 8\}$		3

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$\mathbf{i} \simeq \mathbf{j}$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\,v][j]$	$\{0,1\}$	0
5	$u{\leftarrow}\mathfrak{c}$	?	1
6	$v{\leftarrow}\mathfrak{c}$	?	2
7	$a[i{:=}\,v][j]{\leftarrow}\mathfrak{d}$	?	3
8	$\mathbf{v} \not\simeq \mathbf{a}[\mathbf{i}{:=}\,\mathbf{v}][\mathbf{j}]$	$\{6,7\}$	3
	conflict E^3: $\{2,8\}$		3

Phase 4

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\,v][j]$	$\{0,1\}$	0
5	$v \simeq a[i{:=}\,v][j]$	$\{2\}$	0

An example with arithmetic, arrays, congruence

Phase 3

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$u \simeq v$	$\{\}$	0
4	$u \not\simeq a[i{:=}\,v][j]$	$\{0,1\}$	0
5	$u{\leftarrow}\mathfrak{c}$	?	1
6	$v{\leftarrow}\mathfrak{c}$	?	2
7	$a[i{:=}\,v][j]{\leftarrow}\mathfrak{d}$	?	3
8	$v \not\simeq a[i{:=}\,v][j]$	$\{6,7\}$	3
	conflict E^3: $\{2,8\}$		3

Phase 4

id	trail items	just.	lev.
0	$f(a[i{:=}\,v][j]) \simeq w$	$\{\}$	0
1	$w{-}2 \simeq f(u)$	$\{\}$	0
2	$i \simeq j$	$\{\}$	0
3	$\mathbf{u} \simeq \mathbf{v}$	$\{\}$	0
4	$\mathbf{u} \not\simeq \mathbf{a}[\mathbf{i}{:=}\,\mathbf{v}][\mathbf{j}]$	$\{0,1\}$	0
5	$\mathbf{v} \simeq \mathbf{a}[\mathbf{i}{:=}\,\mathbf{v}][\mathbf{j}]$	$\{2\}$	0
	conflict E^4: $\{3,4,5\}$		0

Example for LRA

LRA-public sorts: just $\mathbb{Q}$.

Example for LRA

LRA-public sorts: just Q. LRA-values: $\mathbb{Q}$. LRA$^+$: trivial

Example for LRA

LRA-public sorts: just Q. LRA-values: $\mathbb{Q}$. LRA^+: trivial

(Some) LRA-inferences:

- Evaluations:

$$t_1{\leftarrow}q_1, \ldots, t_n{\leftarrow}q_n \vdash_{\mathsf{LRA}} l{\leftarrow}b$$

  where l evaluates to b under the assignments

Example for LRA

LRA-public sorts: just Q. LRA-values: $\mathbb{Q}$. LRA$^+$: trivial
(Some) LRA-inferences:

- Evaluations:

$$t_1 \leftarrow q_1, \ldots, t_n \leftarrow q_n \vdash_{\mathsf{LRA}} l \leftarrow b$$

  where l evaluates to b under the assignments

- Fourier-Motzkin resolutions:

$$(e_1 \lessdot_1 x), (x \lessdot_2 e_2) \vdash_{\mathsf{LRA}} (e_1 \lessdot_3 e_2)$$

  where $\lessdot$ is $<$ or $\leq$...
  (triggered only where e_1 and e_2 have been assigned values)

Example for LRA

LRA-public sorts: just Q. LRA-values: $\mathbb{Q}$. LRA$^+$: trivial
(Some) LRA-inferences:

- Evaluations:

$$t_1 \leftarrow q_1, \ldots, t_n \leftarrow q_n \vdash_{\mathsf{LRA}} l \leftarrow b$$

  where l evaluates to b under the assignments

- Fourier-Motzkin resolutions:

$$(e_1 \lessdot_1 x), (x \lessdot_2 e_2) \vdash_{\mathsf{LRA}} (e_1 \lessdot_3 e_2)$$

  where $\lessdot$ is $<$ or $\leq$...
  (triggered only where e_1 and e_2 have been assigned values)

- Treatment of disequality:

$$(e_1 \leq x), (x \leq e_2), (e_1 \simeq e_0), (e_2 \simeq e_0), (x \not\simeq e_0) \vdash_{\mathsf{LRA}} \bot$$

  (triggered only where e_0, e_1 and e_2 have been assigned values)