Recent trends in SMT solving for nonlinear real arithmetic

Gereon Kremer

March 10, 2022
SRI International, Menlo Park

Satisfiability Modulo Theories

$$\exists \overline{x}.\varphi(\overline{x})$$

Is an existential first-order formula satisfiable?

Satisfiability Modulo Theories

$$\exists \overline{x}.\varphi(\overline{x})$$

Is an existential first-order formula satisfiable?

Theories:

- uninterpreted functions
- arrays
- bit-vectors
- floating-point numbers
- arithmetic
- datatypes
- strings
- ...

Satisfiability Modulo Theories

$$\exists \overline{x}.\varphi(\overline{x})$$

Is an existential first-order formula satisfiable?

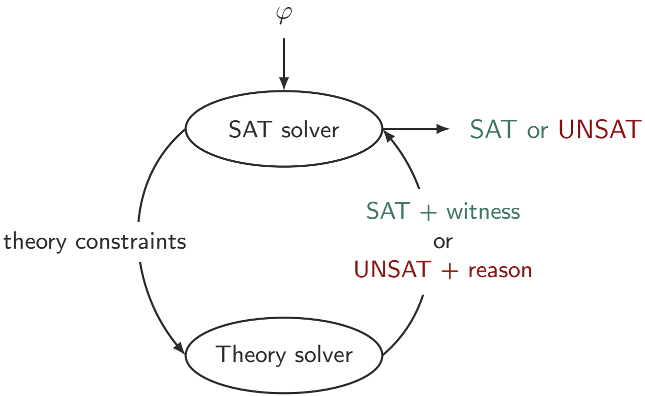
Theories:

- uninterpreted functions
- arrays
- bit-vectors
- floating-point numbers
- arithmetic
- datatypes
- strings
- ...

Extensions:

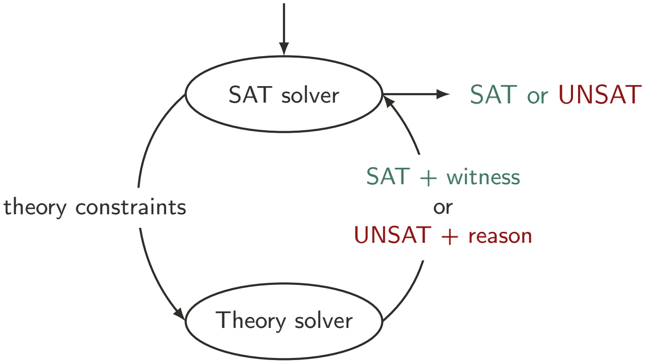
- model generation
- unsat cores
- quantifiers
- optimization queries
- interpolants
- formal proofs
- ...

SMT solving – CDCL(T)

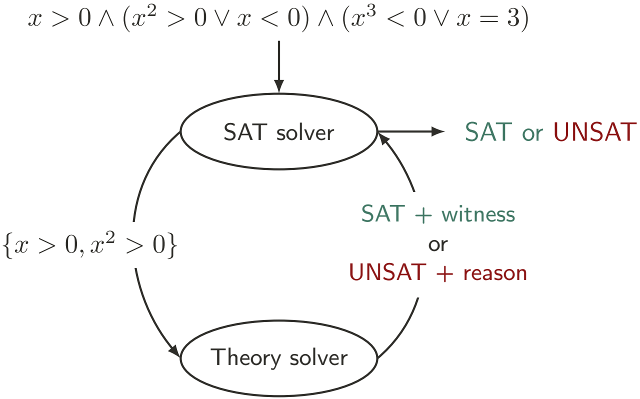


SMT solving – CDCL(T)

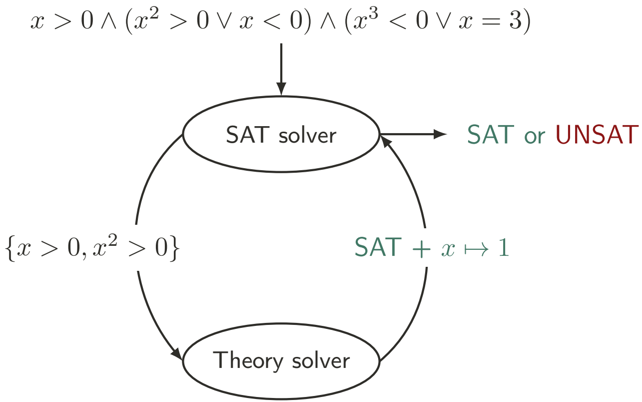
$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3)$$



SMT solving – CDCL(T)

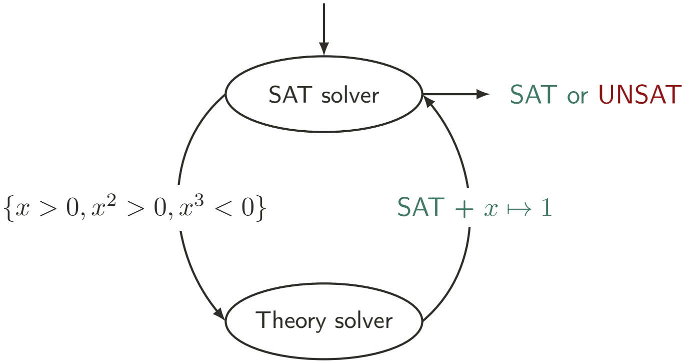


SMT solving – CDCL(T)



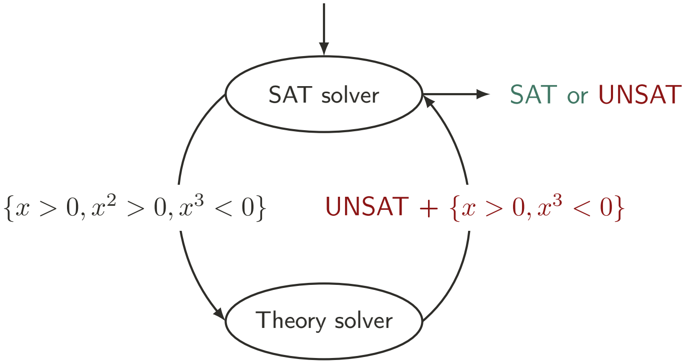
SMT solving – CDCL(T)

$$x > 0 \wedge (x^2 > 0 \vee x < 0) \wedge (x^3 < 0 \vee x = 3)$$



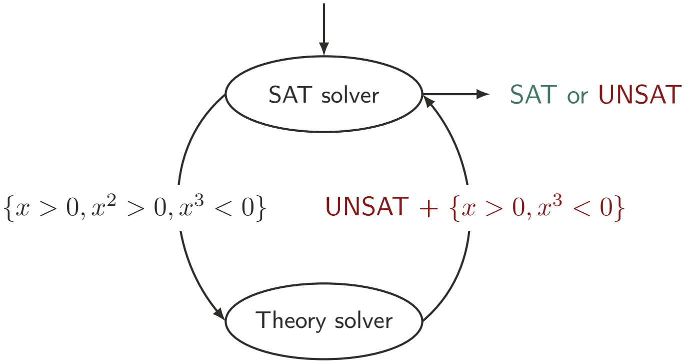
SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3)$$



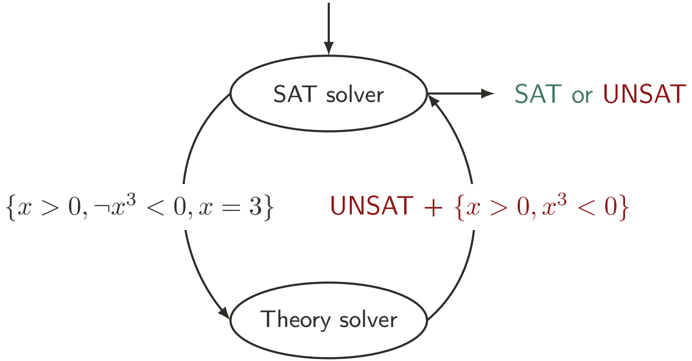
SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3) \land (\neg x > 0 \lor \neg x^3 < 0)$$



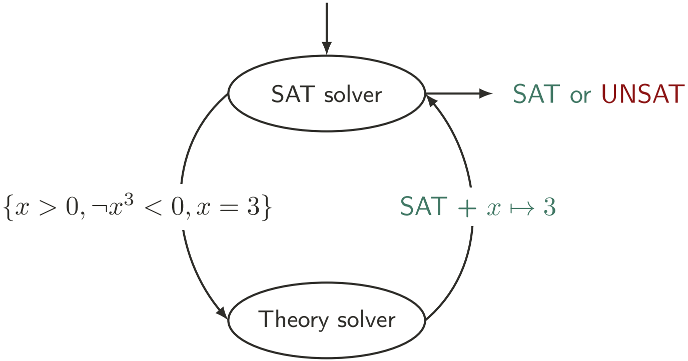
SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3) \land (\neg x > 0 \lor \neg x^3 < 0)$$



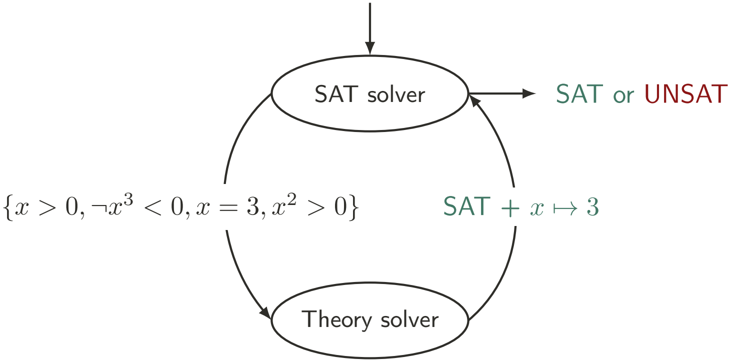
SMT solving – CDCL(T)

$$x > 0 \wedge (x^2 > 0 \vee x < 0) \wedge (x^3 < 0 \vee x = 3) \wedge (\neg x > 0 \vee \neg x^3 < 0)$$

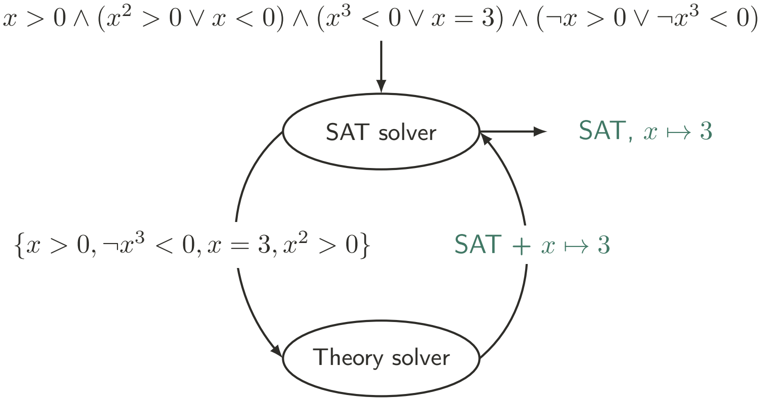


SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3) \land (\neg x > 0 \lor \neg x^3 < 0)$$

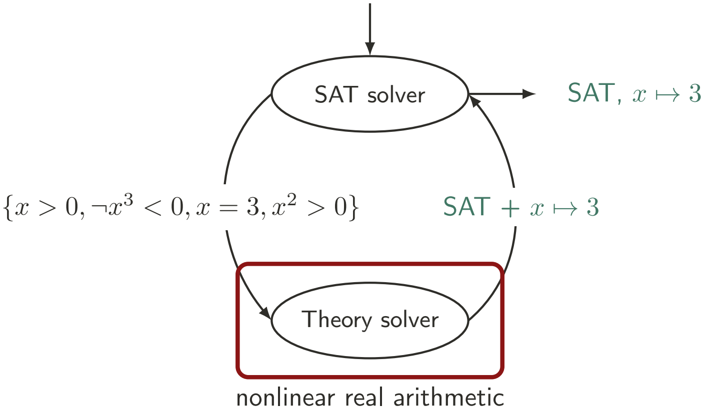


SMT solving – CDCL(T)



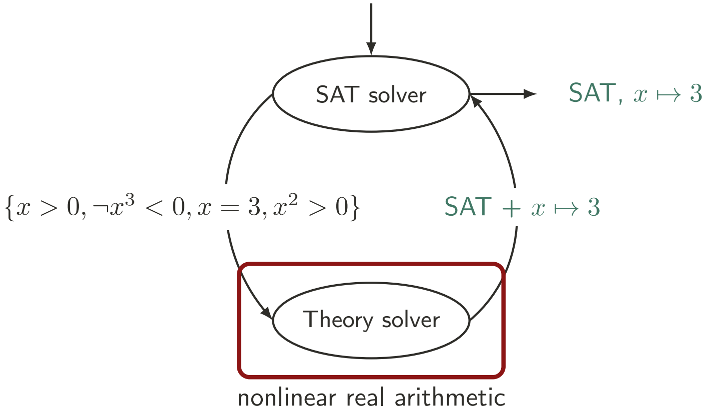
SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3) \land (\neg x > 0 \lor \neg x^3 < 0)$$



SMT solving – CDCL(T)

$$x > 0 \land (x^2 > 0 \lor x < 0) \land (x^3 < 0 \lor x = 3) \land (\neg x > 0 \lor \neg x^3 < 0)$$

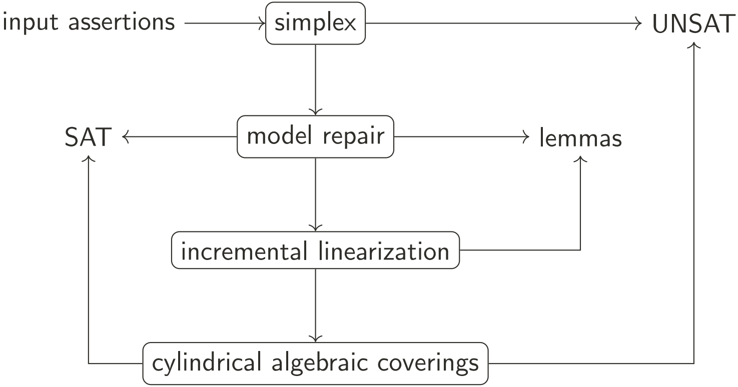


Also: NLSAT/MCSAT [Jovanović et al. 2012] [Moura et al. 2013]

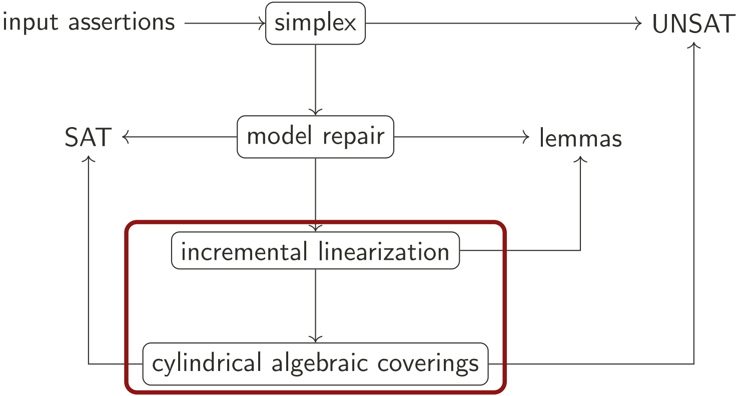
Some theory solvers for real arithmetic

- Simplex
  the go-to method for linear real arithmetic
- Interval Constraint Propagation [Gao et al. 2013]
  shrink search space using interval arithmetic:
  $(0 \leq x \leq 2) \wedge (y = x^2) \Rightarrow (0 \leq y \leq 4)$
- Incremental linearization [Cimatti et al. 2018]
  on-demand lemmas that axiomatize nonlinear functions
- Virtual term substitution [Weispfenning 1997]
  use solution formulas to eliminate variables, only for bounded degrees
- Gröbner basis [Junges 2012]
  canonical characterization of complex solutions, $1 \in GB \Rightarrow$ UNSAT
- Cylindrical Algebraic Decomposition / Coverings [Ábrahám et al. 2021]
  decompose real space into equisatisfiable regions

Arithmetic solving in cvc5



Arithmetic solving in cvc5



Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

$$x > 2 \land y > -1 \land x \cdot y < 2$$

[Cimatti et al. 2018]

Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

$$\begin{aligned} & x > 2 \land y > -1 \land x \cdot y < 2 \\ \text{Model:}\ & x \mapsto 3, y \mapsto 0, x \cdot y \mapsto 1 \\ \text{Lemma:}\ & (x = 0 \lor y = 0) \Rightarrow x \cdot y = 0 \end{aligned}$$

[Cimatti et al. 2018]

Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

$$\begin{aligned} & x > 2 \wedge y > -1 \wedge x \cdot y < 2 \\ \text{Model: } & x \mapsto 3, y \mapsto 0, x \cdot y \mapsto 1 \\ \text{Lemma: } & (x = 0 \vee y = 0) \Rightarrow x \cdot y = 0 \\ \text{Model: } & x \mapsto 3, y \mapsto 1, x \cdot y \mapsto 0 \\ \text{Lemma: } & (x = 3 \wedge y = 1) \Rightarrow x \cdot y = 3 \end{aligned}$$

[Cimatti et al. 2018]

Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

$$
\begin{aligned}
& x > 2 \land y > -1 \land x \cdot y < 2 \\
\text{Model: } & x \mapsto 3, y \mapsto 0, x \cdot y \mapsto 1 \\
\text{Lemma: } & (x = 0 \lor y = 0) \Rightarrow x \cdot y = 0 \\
\text{Model: } & x \mapsto 3, y \mapsto 1, x \cdot y \mapsto 0 \\
\text{Lemma: } & \cancel{(x = 3 \land y = 1) \Rightarrow x \cdot y = 3}
\end{aligned}
$$

[Cimatti et al. 2018]

Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

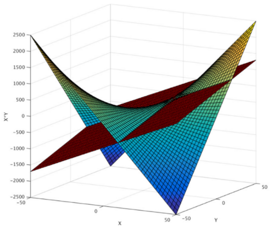
$$x > 2 \wedge y > -1 \wedge x \cdot y < 2$$

$$\text{Model: } x \mapsto 3, y \mapsto 0, x \cdot y \mapsto 1$$

$$\text{Lemma: } (x = 0 \vee y = 0) \Rightarrow x \cdot y = 0$$

$$\text{Model: } x \mapsto 3, y \mapsto 1, x \cdot y \mapsto 0$$

$$\text{Lemma: } \sout{(x = 3 \wedge y = 1) \Rightarrow x \cdot y = 3}$$



[Cimatti et al. 2018]

[Cimatti et al. 2018]

Incremental linearization

implicitly linearize: $x \cdot y \rightsquigarrow a_{x \cdot y}$

$$x > 2 \land y > -1 \land x \cdot y < 2$$

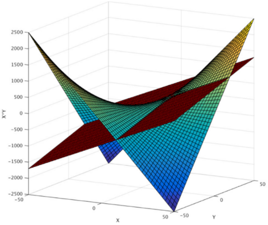
Model: $x \mapsto 3, y \mapsto 0, x \cdot y \mapsto 1$

Lemma: $(x = 0 \lor y = 0) \Rightarrow x \cdot y = 0$

Model: $x \mapsto 3, y \mapsto 1, x \cdot y \mapsto 0$

Lemma: ~~$(x = 3 \land y = 1) \Rightarrow x \cdot y = 3$~~

$$(x \leq 3 \land y \leq 1) \lor (x \geq 3 \land y \geq 1)$$
$$\Leftrightarrow (x \cdot y \geq 1 \cdot x + 3 \cdot y - 3 \cdot 1)$$



[Cimatti et al. 2018]

[Cimatti et al. 2018]

Cylindrical Algebraic Coverings

- Guess partial assignment

$$s_1 \times \cdots \times s_k \times s_{k+1}$$

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings

- Guess partial assignment

$$s_1 \times \cdots \times s_k \times s_{k+1}$$

- Refute partial assignment using intervals

$$s \notin s_1 \times \cdots \times s_k \times (a, b)$$

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings

- Guess partial assignment

$$s_1 \times \cdots \times s_k \times s_{k+1}$$

- Refute partial assignment using intervals

$$s \notin s_1 \times \cdots \times s_k \times (a, b)$$

- Lift covering to lower dimension

$$s_1 \times \cdots \times s_k \times \{(-\infty, a), [a, b], \dots (z, \infty)\} \rightarrow s_1 \times \cdots \times s_{k-1} \times (\alpha, \beta)$$

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings

- Guess partial assignment

$$s_1 \times \cdots \times s_k \times s_{k+1}$$

- Refute partial assignment using intervals

$$s \notin s_1 \times \cdots \times s_k \times (a, b)$$

- Lift covering to lower dimension

$$s_1 \times \cdots \times s_k \times \{(-\infty, a), [a, b], \ldots (z, \infty)\} \rightarrow s_1 \times \cdots \times s_{k-1} \times (\alpha, \beta)$$

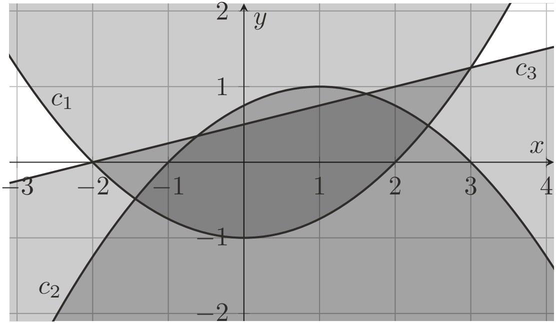
- Eventually get satisfying assignment or a covering in first dimension

$$s = s_1 \times \cdots \times s_n \qquad \text{or} \qquad s_1 \notin \{(-\infty, a), [a, b], \ldots (z, \infty)\}$$

[Ábrahám et al. 2021] [Kremer et al. 2021]

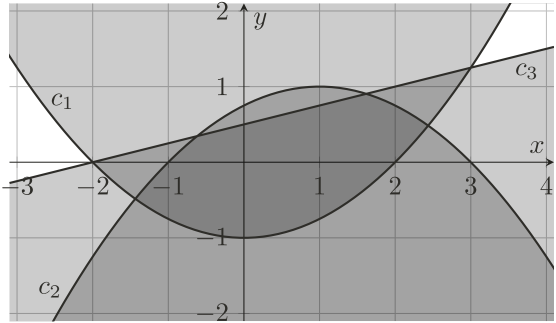
An example

$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



An example

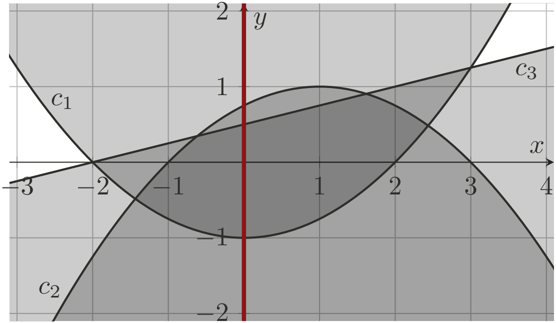
$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



No constraint for x

An example

$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$

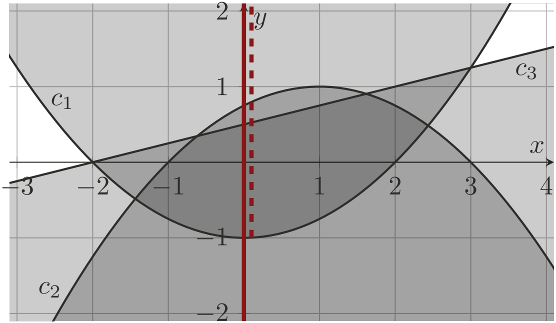


No constraint for x

Guess $x \mapsto 0$

An example

$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



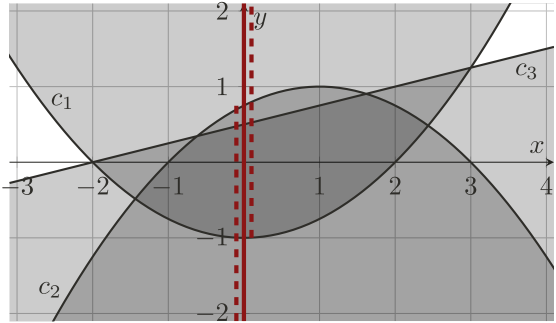
No constraint for x

Guess $x \mapsto 0$

$c_1 \to y \notin (-1, \infty)$

An example

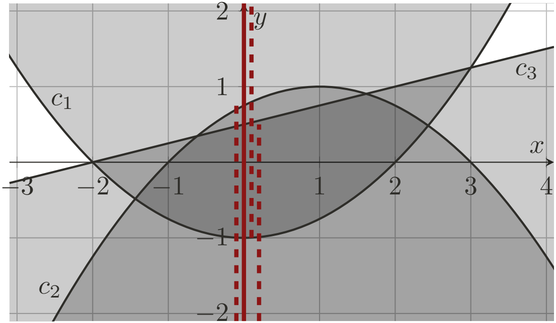
$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$



No constraint for x
Guess $x \mapsto 0$
$c_1 \to y \notin (-1, \infty)$
$c_2 \to y \notin (-\infty, 0.75)$

An example

$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



No constraint for x

Guess $x \mapsto 0$

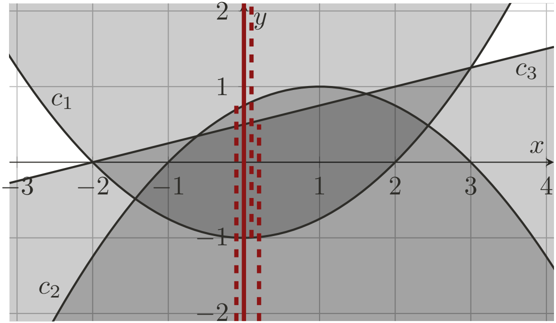
$c_1 \to y \notin (-1, \infty)$

$c_2 \to y \notin (-\infty, 0.75)$

$c_3 \to y \notin (-\infty, 0.5)$

An example

$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$



No constraint for x

Guess $x \mapsto 0$

$c_1 \to y \notin (-1, \infty)$

$c_2 \to y \notin (-\infty, 0.75)$

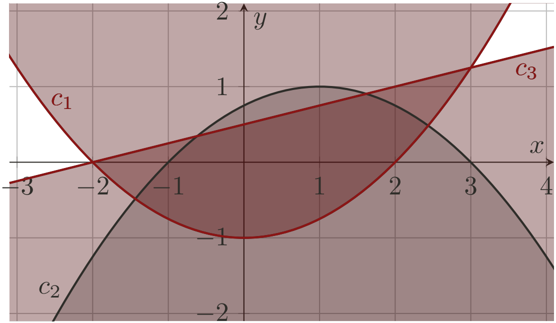
$c_3 \to y \notin (-\infty, 0.5)$

Construct covering

$(-\infty, 0.5), (-1, \infty)$

An example

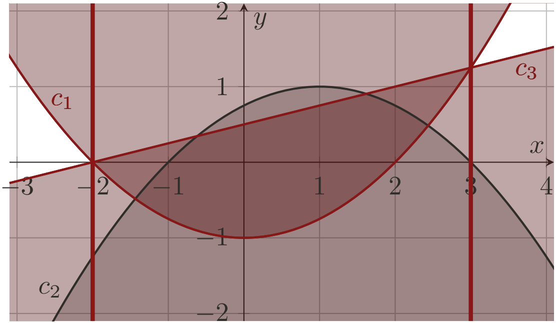
$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



No constraint for x
Guess $x \mapsto 0$
$c_1 \to y \notin (-1, \infty)$
$c_2 \to y \notin (-\infty, 0.75)$
$c_3 \to y \notin (-\infty, 0.5)$
Construct covering
$(-\infty, 0.5), (-1, \infty)$

An example

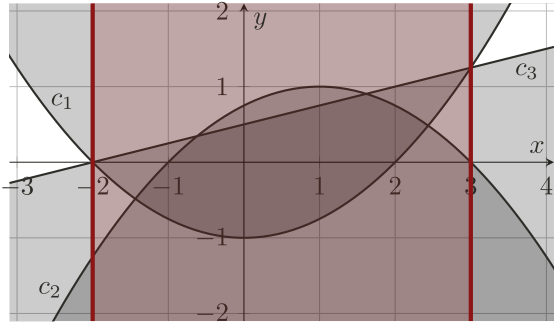
$$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$$



No constraint for x
Guess $x \mapsto 0$
$c_1 \to y \notin (-1, \infty)$
$c_2 \to y \notin (-\infty, 0.75)$
$c_3 \to y \notin (-\infty, 0.5)$
Construct covering
$(-\infty, 0.5), (-1, \infty)$
Construct interval for x
$x \notin (-2, 3)$

An example

$c_1 : 4 \cdot y < x^2 - 4 \qquad c_2 : 4 \cdot y > 4 - (x-1)^2 \qquad c_3 : 4 \cdot y > x + 2$



No constraint for x
Guess $x \mapsto 0$
$c_1 \to y \notin (-1, \infty)$
$c_2 \to y \notin (-\infty, 0.75)$
$c_3 \to y \notin (-\infty, 0.5)$
Construct covering
$(-\infty, 0.5), (-1, \infty)$
Construct interval for x
$x \notin (-2, 3)$
New guess for x

Cylindrical Algebraic Coverings – main algorithm

function get_unsat_cover$((s_1, \ldots, s_{i-1}))$

$\mathbb{I}$:= get_unsat_intervals(s)
while $\bigcup_{I \in \mathbb{I}} I \neq \mathbb{R}$ **do**
  s_i := sample_outside$(\mathbb{I})$
  if $i = n$ **then** **return** $(\mathrm{SAT}, (s_1, \ldots, s_{i-1}, s_i))$
  (f, O) := get_unsat_cover$((s_1, \ldots, s_{i-1}, s_i))$
  if $f = \mathrm{SAT}$ **then** **return** (SAT, O)
  else if $f = \mathrm{UNSAT}$ **then**
    R := construct_characterization$((s_1, \ldots, s_{i-1}, s_i), O)$
    J := interval_from_characterization$((s_1, \ldots, s_{i-1}), s_i, R)$
    $\mathbb{I} := \mathbb{I} \cup \{J\}$
return $(\mathrm{UNSAT}, \mathbb{I})$

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

function get_unsat_cover($(s_1, \ldots, s_{i-1})$)

$\mathbb{I}$:= get_unsat_intervals(s)

while $\bigcup_{I \in \mathbb{I}} I \neq \mathbb{R}$ **do**

  s_i := sample_outside($\mathbb{I}$)

  if $i = n$ **then** **return** $(\text{SAT}, (s_1, \ldots, s_{i-1}, s_i))$

  (f, O) := get_unsat_cover($(s_1, \ldots, s_{i-1}, s_i)$)

  if $f = \text{SAT}$ **then** **return** (SAT, O)

  else if $f = \text{UNSAT}$ **then**

    R := construct_characterization($(s_1, \ldots, s_{i-1}, s_i), O$)

    J := interval_from_characterization($(s_1, \ldots, s_{i-1}), s_i, R$)

    $\mathbb{I} := \mathbb{I} \cup \{J\}$

return $(\text{UNSAT}, \mathbb{I})$

Real root isolation over a partial sample point

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

```
function get_unsat_cover((s_1, ..., s_{i-1}))

𝕀 := get_unsat_intervals(s)
while ⋃_{I∈𝕀} I ≠ ℝ do
  s_i := sample_outside(𝕀)
  if i = n then  return (SAT, (s_1, ..., s_{i-1}, s_i))
  (f, O) := get_unsat_cover((s_1, ..., s_{i-1}, s_i))
  if f = SAT then  return (SAT, O)
  else if f = UNSAT then
    R := construct_characterization((s_1, ..., s_{i-1}, s_i), O)
    J := interval_from_characterization((s_1, ..., s_{i-1}), s_i, R)
    𝕀 := 𝕀 ∪ {J}
return (UNSAT, 𝕀)
```

Real root isolation over a partial sample point

Select sample from $\mathbb{R} \setminus I$

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

`function get_unsat_cover`$((s_1, \ldots, s_{i-1}))$

$\mathbb{I} :=$ `get_unsat_intervals(s)`
while $\bigcup_{I \in \mathbb{I}} I \neq \mathbb{R}$ **do**
  $s_i :=$ `sample_outside`$(\mathbb{I})$
  if $i = n$ **then** **return** $(\text{SAT}, (s_1, \ldots, s_{i-1}, s_i))$
  $(f, O) :=$ `get_unsat_cover`$((s_1, \ldots, s_{i-1}, s_i))$
  if $f = \text{SAT}$ **then** **return** (SAT, O)
  else if $f = \text{UNSAT}$ **then**
    $R :=$ `construct_characterization`$((s_1, \ldots, s_{i-1}, s_i), O)$
    $J :=$ `interval_from_characterization`$((s_1, \ldots, s_{i-1}), s_i, R)$
    $\mathbb{I} := \mathbb{I} \cup \{J\}$
return $(\text{UNSAT}, \mathbb{I})$

Real root isolation over a partial sample point

Select sample from $\mathbb{R} \setminus I$

Recurse to next variable

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

function get_unsat_cover($(s_1, \ldots, s_{i-1})$)

$\mathbb{I} :=$ get_unsat_intervals(s)

while $\bigcup_{I \in \mathbb{I}} I \neq \mathbb{R}$ **do**

$s_i :=$ sample_outside($\mathbb{I}$)

if $i = n$ **then return** $(\mathrm{SAT}, (s_1, \ldots, s_{i-1}, s_i))$

$(f, O) :=$ get_unsat_cover($(s_1, \ldots, s_{i-1}, s_i)$)

if $f = \mathrm{SAT}$ **then return** (SAT, O)

else if $f = \mathrm{UNSAT}$ **then**

$R :=$ construct_characterization($(s_1, \ldots, s$

$J :=$ interval_from_characterization($(s_1,$

$\mathbb{I} := \mathbb{I} \cup \{J\}$

return $(\mathrm{UNSAT}, \mathbb{I})$

Real root isolation over a partial sample point

Select sample from $\mathbb{R} \setminus I$

Recurse to next variable

CAD-style projection: Roots of polynomials restrict where covering is still applicable

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

function get_unsat_cover($(s_1, \ldots, s_{i-1})$)

$\mathbb{I} :=$ get_unsat_intervals(s)
while $\bigcup_{I \in \mathbb{I}} I \neq \mathbb{R}$ **do**
  $s_i :=$ sample_outside($\mathbb{I}$)
  if $i = n$ **then return** $(\mathrm{SAT}, (s_1, \ldots, s_{i-1}, s_i))$
  $(f, O) :=$ get_unsat_cover($(s_1, \ldots, s_{i-1}, s_i)$)
  if $f = \mathrm{SAT}$ **then return** (SAT, O)
  else if $f = \mathrm{UNSAT}$ **then**
    $R :=$ construct_characterization($(s_1, \ldots, s$
    $J :=$ interval_from_characterization($(s_1,$
    $\mathbb{I} := \mathbb{I} \cup \{J\}$
return $(\mathrm{UNSAT}, \mathbb{I})$

Real root isolation over a partial sample point

Select sample from $\mathbb{R} \setminus I$

Recurse to next variable

CAD-style projection: Roots of polynomials restrict where covering is still applicable

Extract interval from polynomials

[Ábrahám et al. 2021] [Kremer et al. 2021]

Cylindrical Algebraic Coverings – main algorithm

```
function get_unsat_cover((s_1, ..., s_{i-1}))

𝕀 := get_unsat_intervals(s)
while ⋃_{I∈𝕀} I ≠ ℝ do
  s_i := sample_outside(𝕀)
  if i = n then  return (SAT, (s_1, ..., s_{i-1}, s_i))
  (f, O) := get_unsat_cover((s_1, ..., s_{i-1}, s_i))
  if f = SAT then  return (SAT, O)
  else if f = UNSAT then
    R := construct_characterization((s_1, ..., s
    J := interval_from_characterization((s_1,
    𝕀 := 𝕀 ∪ {J}
return (UNSAT, 𝕀)
```

Real root isolation over a partial sample point

Select sample from $\mathbb{R} \setminus I$

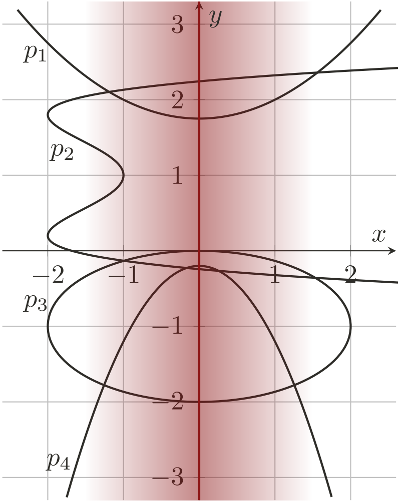
Recurse to next variable

CAD-style projection: Roots of polynomials restrict where covering is still applicable

Extract interval from polynomials

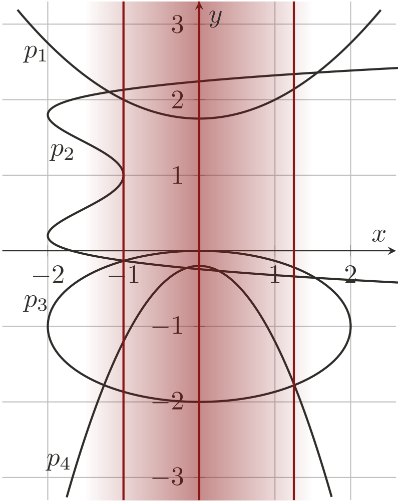
[Ábrahám et al. 2021] [Kremer et al. 2021]

construct_characterization



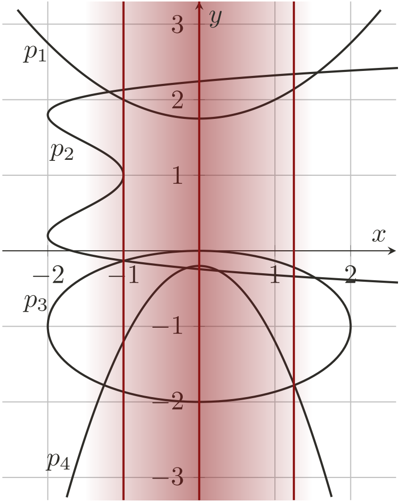
Identify region around sample

construct_characterization



Identify region around sample

construct_characterization

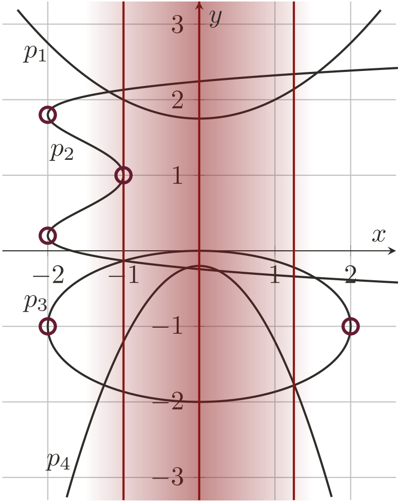


Identify region around sample

CAD projection:

- Discriminants (and coefficients)
- Resultants

construct_characterization

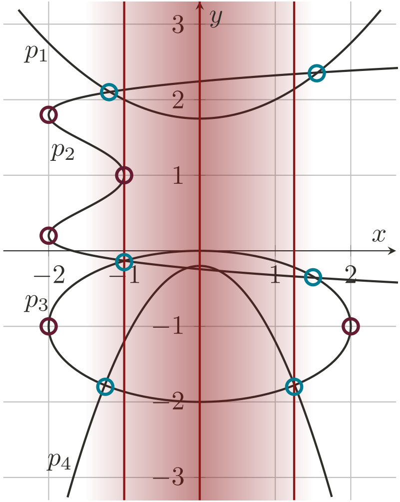


Identify region around sample

CAD projection:

- Discriminants (and coefficients)
- Resultants

construct_characterization

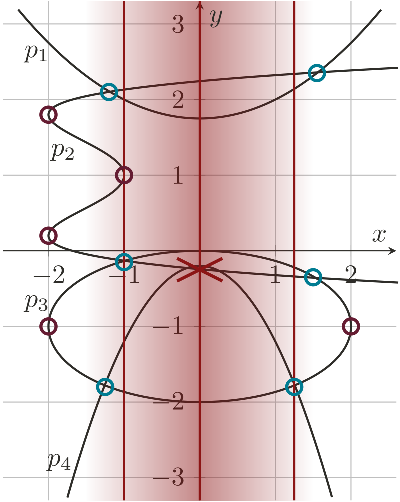


Identify region around sample

CAD projection:

- Discriminants (and coefficients)
- Resultants

construct_characterization



Identify region around sample

CAD projection:

- Discriminants (and coefficients)
- Resultants

Improvement over CAD:

- Resultants between neighbouring intervals only!

On the implementation in cvc5

- Heavily based on LibPoly [Jovanovic et al. 2017]
- Implements Lazard's lifting [Lazard 1994] [Kremer et al. 2021]
  using CoCoALib [Abbott et al. 2018]
- Different variable orderings inspired by [England et al. 2014]
  Nothing spectacular, though
- Generates infeasible subsets
- Allows for partial checks
  Not useful in our context as lemmas are nonlinear
- Supports mixed-integer problems
- Experimental support for incremental checks
  No performance benefit observed
- Generation of formal proof skeletons
  Helps understanding, not detailed enough for automated verification
- Arbitrary theory combination
  Real algebraic numbers are first-class citizens of cvc5

Experiments – QF_NRA

QF_NRA	sat	unsat	solved
cvc5	**5137**	**5596**	**10733**
Yices2	4966	5450	10416
z3	5136	5207	10343
cvc5.cov	5001	5077	10078
SMT-RAT	4828	5038	9866
veriT	4522	5034	9556
MathSAT	3645	5357	9002
cvc5.inclin	3421	5376	8797

Experiments – NRA & QF_UFNRA

Beyond QF_NRA		sat	unsat	solved
NRA	Yices2	231	**3817**	**4048**
	z3	**236**	3812	**4048**
	cvc5.cov	**236**	3809	4045
	cvc5	221	3809	4030
	cvc5.inclin	120	3786	3906
QF_UFNRA	z3	**24**	**11**	**35**
	Yices2	23	**11**	34
	cvc5	20	**11**	31
	cvc5.inclin	12	**11**	23
	cvc5.cov	2	**11**	13

Thank you for your attention!
Any questions?

References I

- John Abbott, Anna M. Bigatti, and Elisa Palezzato. "New in CoCoA-5.2.4 and CoCoALib-0.99600 for SC-Square". In: SC^2. FLoC. July 2018. URL: http://ceur-ws.org/Vol-2189/paper4.pdf.
- Erika Ábrahám, James H. Davenport, Matthew England, and Gereon Kremer. "Deciding the Consistency of Non-Linear Real Arithmetic Constraints with a Conflict Driven Search Using Cylindrical Algebraic Coverings". In: Journal of Logical and Algebraic Methods in Programming 119 (2021), p. 100633. DOI: 10.1016/j.jlamp.2020.100633.
- Alessandro Cimatti, Alberto Griggio, Ahmed Irfan, Marco Roveri, and Roberto Sebastiani. "Incremental Linearization for Satisfiability and Verification Modulo Nonlinear Arithmetic and Transcendental Functions". In: ACM Transactions on Computational Logic 19 (3 2018), 19:1–19:52. DOI: 10.1145/3230639.
- Matthew England, Russell Bradford, James H. Davenport, and David Wilson. "Choosing a Variable Ordering for Truth-Table Invariant Cylindrical Algebraic Decomposition by Incremental Triangular Decomposition". In: ICMS. 2014. DOI: 10.1007/978-3-662-44199-2_68.
- Sicun Gao, Soonho Kong, and Edmund M. Clarke. "dReal: An SMT Solver for Nonlinear Theories over the Reals". In: CADE-24. 2013. DOI: 10.1007/978-3-642-38574-2_14.
- Dejan Jovanovic and Bruno Dutertre. "LibPoly: A Library for Reasoning about Polynomials". In: SMT. CAV. 2017. URL: http://ceur-ws.org/Vol-1889/paper3.pdf.
- Dejan Jovanović and Leonardo de Moura. "Solving Non-linear Arithmetic". In: IJCAR. 2012. DOI: 10.1007/978-3-642-31365-3_27.
- Sebastian Junges. "On Gröbner Bases in SMT-Compliant Decision Procedures". Bachelor's thesis. RWTH Aachen University, 2012.

References II

- Gereon Kremer, Erika Ábrahám, Matthew England, and James H. Davenport. "On the Implementation of Cylindrical Algebraic Coverings for Satisfiability Modulo Theories Solving". In: SYNASC. 2021. DOI: 10.1109/SYNASC54541.2021.00018.
- Gereon Kremer and Jens Brandt. "Implementing arithmetic over algebraic numbers A tutorial for Lazard's lifting scheme in CAD". In: SYNASC. 2021. DOI: 10.1109/SYNASC54541.2021.00013.
- Daniel Lazard. "An Improved Projection for Cylindrical Algebraic Decomposition". In: Algebraic Geometry and its Applications. 1994. Chap. 29, pp. 467–476. DOI: 10.1007/978-1-4612-2628-4_29.
- Leonardo de Moura and Dejan Jovanović. "A Model-Constructing Satisfiability Calculus". In: VMCAI. 2013. DOI: 10.1007/978-3-642-35873-9_1.
- Volker Weispfenning. "Quantifier Elimination for Real Algebra — the Quadratic Case and Beyond". In: Applicable Algebra in Engineering, Communication and Computing 8 (2 1997), pp. 85–101. DOI: 10.1007/s002000050055.