Solving non-linear constraints CDCL-style [1]

F. Brauße⋆ K. Korovin□ M. Korovina⋄ N. Th. Müller⋆

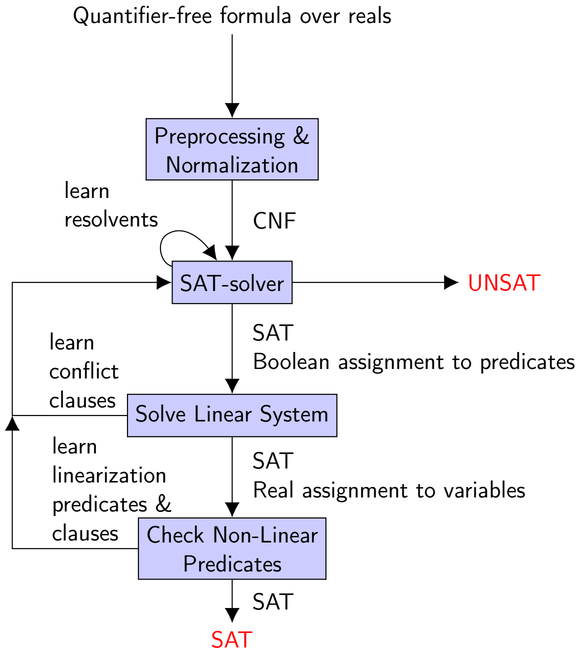
⋆Universität Trier

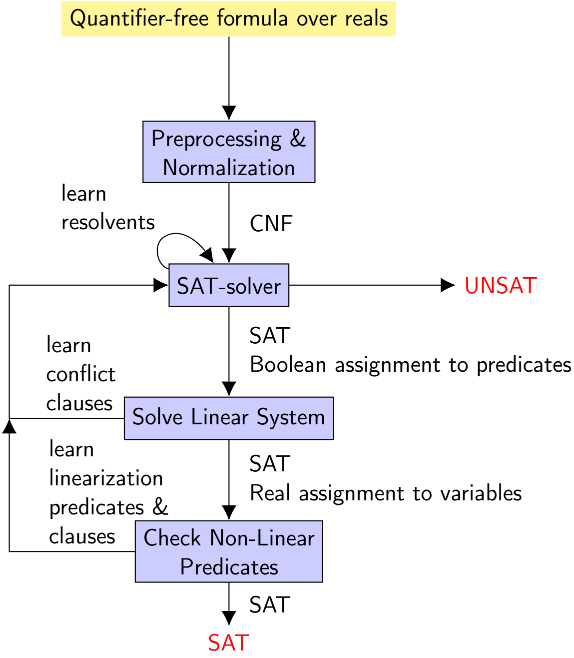
□University of Manchester

⋄Ershov Inst., Novosibirsk

Workshop on MPFR/MPC/iRRAM, Trier, 2018-11-22

[1] This project has received funding from the DFG grant WERA MU 1801/5-1, the DFG/RFBR grant CAVER BE 1267/14-1 and 14-01-91334 and from the EU's Horizon 2020 programme under the Marie Skłodowska-Curie grant agreement No 731143.


Quantifier-free formula over reals
Preprocessing & Normalization
learn resolvents
CNF
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
learn linearization predicates & clauses
SAT
Real assignment to variables
Check Non-Linear Predicates
SAT
SAT


Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

```
(set-info :smt-lib-version 2.6)
(set-logic QF_NRA)
(set-info :source |
These benchmarks used in the paper:

  Dejan Jovanovic and Leonardo de Moura.  Solving Non-Linear Arithmetic.
  In IJCAR 2012, published as LNCS volume 7364, pp. 339--354.

The meti-tarski benchmarks are proof obligations extracted from the
Meti-Tarski project, see:

  B. Akbarpour and L. C. Paulson. MetiTarski: An automatic theorem prover
  for real-valued special functions. Journal of Automated Reasoning,
  44(3):175-205, 2010.

Submitted by Dejan Jovanovic for SMT-LIB.
|)

(set-info :category "industrial")
(set-info :status sat)

(declare-fun skoY () Real)
(declare-fun skoX () Real)
(declare-fun skoZ () Real)

(assert (and (not (<= (* skoZ (+ (+ (* skoX (/ 213 1000)) (* skoY (/ 413 10000)))
                                 (* skoZ (/ (- 18) 25))))
                      (+ (+ (/ (- 1) 10) (* skoX (* skoX (/ 261 100))))
                         (* skoY (+ (* skoX (/ 21 20)) (* skoY (/ 141 100))))))))
             (or (not (= skoX 0))
                 (or (not (= skoY 0))
                     (not (= skoZ 0))))))

(check-sat)
(exit)
```

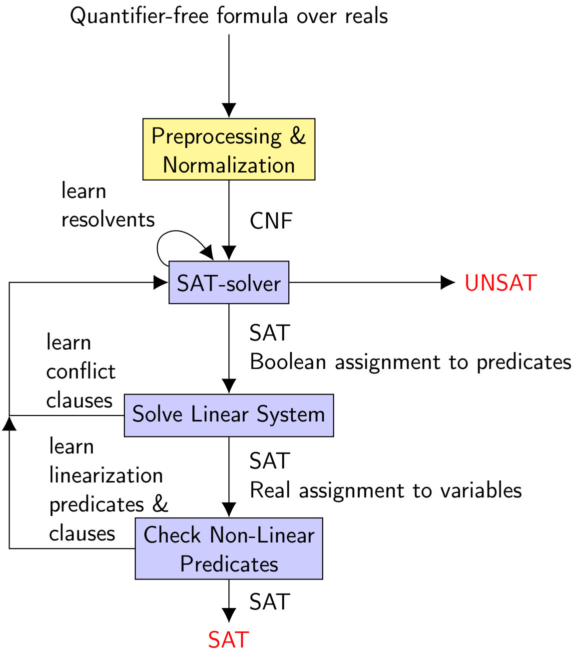
... more readable as

$$\exists x, y, z \in \mathbb{R}_{\mathsf{alg}} : \neg\Big(z \cdot (x \cdot \tfrac{213}{1000} + y \cdot \tfrac{413}{10000} + z \cdot \tfrac{-18}{25}) \leq \tfrac{-1}{10} + x \cdot x \cdot \tfrac{261}{100} + y \cdot (x \cdot \tfrac{21}{20} + y \cdot \tfrac{141}{100})\Big) \wedge (\neg(x = 0) \vee \neg(y = 0) \vee \neg(z = 0))$$

Satisfiability modulo algebraic theory of real numbers.

Many solvers exist (also for other theories). Popular:

- `z3` (Microsoft Research)
- `cvc4` (Stanford Univ.)
- ...see `http://smtcomp.sourceforge.net/2018/participants.shtml`


Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

- Tseytin transformation
- Total order on terms
- Field axioms
- Introducing new boolean/real variables

Transform to CNF

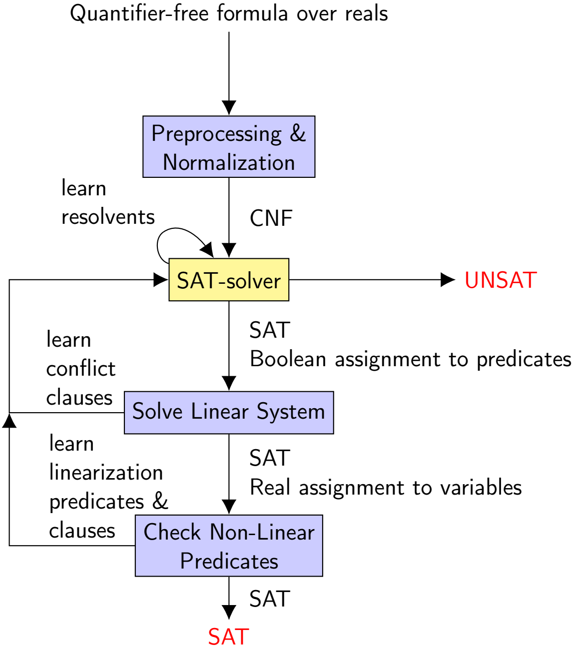
$$\bigwedge_i \bigvee_j \ell_{ij}$$

where ℓ_{ij} are literals of boolean variables or of predicates in normalized form

$$\underbrace{c_0}_{\text{constant}} + \underbrace{\sum c_x x}_{\text{linear}} + \underbrace{cf(x)}_{\text{non-linear}} \;\theta\; 0$$

where $\theta \in \{\neq, =, <, \leq, >, \geq\}$, $f \in \mathcal{F}$, $c_0, c_x, c \in \mathbb{Q}$.

Currently implemented $\mathcal{F} = \{x \mapsto x^n, \mathrm{abs} : n \in \mathbb{N}\}$.


Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

Given CNF $\bigwedge_i \bigvee_j \ell_{ij}$, either shows SAT by finding an assignment

$$\beta : \{v_1, \dots, v_n\} \to \{0, 1\}$$

or proves UNSAT by deriving the empty clause via resolution:

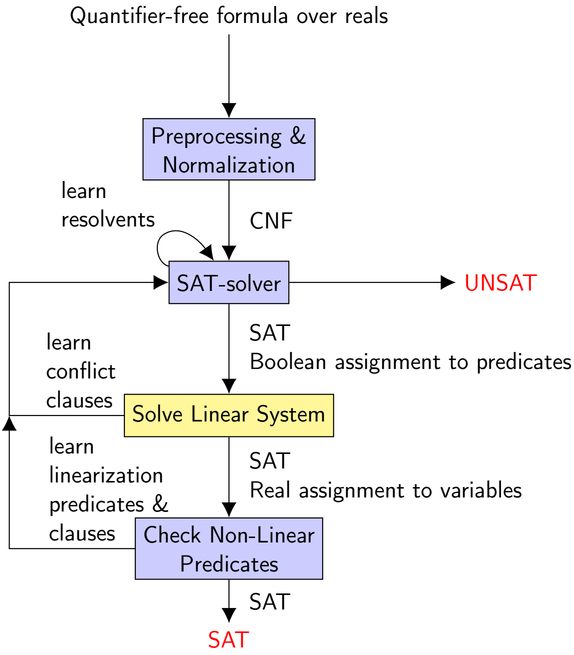
$$\frac{(a_1 \vee \cdots \vee a_n \vee \ell), (b_1 \vee \cdots \vee b_m \vee \neg\ell)}{(a_1 \vee \cdots \vee a_n \vee b_1 \vee \cdots \vee b_m)}$$

Algorithms:

- DP '60: generate all resolutions
- DPLL '62: recursively check every assignment
- CDCL mid '90: learn only resolvents of conflicts derived from partial assignments (popular solvers: `minisat`, ..., `ksat`)

Conflict

$(\phi \Longrightarrow \ell) \wedge (\psi \Longrightarrow \neg\ell)$ where ϕ and ψ are $\top$ under $\beta \rightsquigarrow$ learn clause $\neg(\phi \wedge \psi)$.


Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

Given system of linear predicates $\sum_i c_i x_i \ \theta \ c$, either shows SAT by finding assignment

$$\alpha : \{x_1, \ldots, x_m\} \to \mathbb{R}_{\mathsf{alg}}$$

or shows UNSAT by generating a conflict clause of predicates p_i shown to be a tautology

$$(\neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_k)$$

Algorithms:

- Fourier-Motzkin (Fourier, 1862): generate all resolvents
  - $\rightsquigarrow$ exponential complexity
- Simplex (Dantzig 1961): search all vertices of polytope
  - used by e.g. `z3`
- ellipsoid and interior point (Khachiyan '79; Karmakar '84)

Notice similarities to SAT:

propositional	linear arithmetic
variables v_i	variables x_i
literals ℓ_i	monomials $c_i x_i$
clauses $(\ell_1 \vee \ell_2 \vee \cdots \vee \ell_k)$	linear pred. $c_1 x_1 + c_2 x_2 + \cdots + c_k x_k \;\theta\; c$
clause resolution $\dfrac{\ell \vee C \quad \neg\ell \vee D}{C \vee D}$	linear resolution $\dfrac{-ax + p \leq 0 \quad bx + q \leq 0 \quad a > 0 \leftrightarrow b > 0}{bp + aq \leq 0}$

Adaptation of Fourier-Motzkin method by Korovin, Voronkov 2009/11:

- similar to CDCL: only conflicts drive resolution
- use bound propagation to assign values from narrower intervals

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable				
bounds				
assignment				

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1			
bounds				
assignment				

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1			
bounds	$(-\infty, \infty)$			
assignment				

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1			
bounds	$(-\infty, \infty)$			
assignment	0			

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2		
bounds	$(-\infty, \infty)$			
assignment	0			

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2		
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$		
assignment	0			

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2		
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$		
assignment	0	0		

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	
assignment	0	0	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$
assignment	0	0	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$
assignment	0	0	4

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$	
assignment	0	0	4	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$	$[3; -3]$
assignment	0	0	4	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$	$[3; -3]$
assignment	0	0	4	

Conflict: $\mathrm{res}_{x_4}((1), (3)) \rightsquigarrow (7)$

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 5]$	$[3; -3]$
assignment	0	0	4	

Conflict: $\text{res}_{x_4}((1), (3)) \rightsquigarrow (7)$

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	
assignment	0	0	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 2]$
assignment	0	0	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 2]$
assignment	0	0	1

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1) \\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2) \\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3) \\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4) \\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5) \\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6) \\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 2]$	
assignment	0	0	1	

Example

$$\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 2]$	$[-3; 0]$
assignment	0	0	1	

Example

$$
\begin{array}{rcrcrcrcrcl}
x_4 & - & 2x_3 & & & + & x_1 & + & 5 & \geq & 0 \quad (1)\\
x_4 & + & 2x_3 & + & x_2 & & & + & 3 & \geq & 0 \quad (2)\\
-x_4 & - & x_3 & - & 3x_2 & - & 3x_1 & + & 1 & \geq & 0 \quad (3)\\
-x_4 & + & 2x_3 & + & 2x_2 & + & x_1 & + & 6 & \geq & 0 \quad (4)\\
 & & x_3 & & & + & 3x_1 & - & 1 & \geq & 0 \quad (5)\\
 & & -x_3 & + & x_2 & - & 2x_1 & + & 5 & \geq & 0 \quad (6)\\
 & & -x_3 & - & x_2 & - & \frac{2}{3}x_1 & + & 2 & \geq & 0 \quad (7)
\end{array}
$$

variable	x_1	x_2	x_3	x_4
bounds	$(-\infty, \infty)$	$(-\infty, \infty)$	$[1; 2]$	$[-3; 0]$
assignment	0	0	1	-1

SAT

UNSAT Example

$$\begin{array}{rrrrcll} -y & + & x & & \geq & 0 & (p_1) \qquad p_1 \\ y & & & - \;\; 3 & \geq & 0 & (p_2) \qquad p_2 \\ & & -x & + \;\; 1 & \geq & 0 & (p_3) \qquad p_3 \end{array}$$

variable			
bounds			
assignment			

UNSAT Example

$$\begin{array}{ccccccccc} -y & + & x & & & \geq & 0 & (p_1) & p_1 \\ y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\ & & -x & + & 1 & \geq & 0 & (p_3) & p_3 \end{array}$$

variable	x		
bounds			
assignment			

UNSAT Example

$$
\begin{array}{rrrrcrl}
-y & + & x & & \geq & 0 & (p_1) \qquad p_1 \\
y & & & - \; 3 & \geq & 0 & (p_2) \qquad p_2 \\
& & -x & + \; 1 & \geq & 0 & (p_3) \qquad p_3
\end{array}
$$

variable	x	
bounds	$(-\infty, 1]$	
assignment		

UNSAT Example

$$\begin{array}{rrrrrrl} -y & + & x & & & \geq & 0 \quad (p_1) \qquad p_1 \\ y & & & - & 3 & \geq & 0 \quad (p_2) \qquad p_2 \\ & & -x & + & 1 & \geq & 0 \quad (p_3) \qquad p_3 \end{array}$$

variable	x	
bounds	$(-\infty, 1]$	
assignment	1	

UNSAT Example

$$
\begin{array}{rrrrrrl}
-y & + & x & & & \geq & 0 \quad (p_1) \qquad p_1 \\
y & & & - & 3 & \geq & 0 \quad (p_2) \qquad p_2 \\
 & & -x & + & 1 & \geq & 0 \quad (p_3) \qquad p_3
\end{array}
$$

variable	x	y
bounds	$(-\infty, 1]$	
assignment	1	

UNSAT Example

$$\begin{array}{ccccccccc} -y & + & x & & & \geq & 0 & (p_1) & p_1 \\ y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\ & & -x & + & 1 & \geq & 0 & (p_3) & p_3 \end{array}$$

variable	x	y
bounds	$(-\infty, 1]$	$[3; 1]$
assignment	1	

Conflict: $\mathrm{res}_{x_4}(p_1, p_2) \rightsquigarrow p_4$ with *reason* $p_1 \wedge p_2$

UNSAT Example

$$
\begin{array}{ccccccccc}
-y & + & x & & & \geq & 0 & (p_1) & p_1 \\
y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\
 & & -x & + & 1 & \geq & 0 & (p_3) & p_3 \\
 & & x & - & 3 & \geq & 0 & (p_4) & p_1 \wedge p_2
\end{array}
$$

variable	x	y
bounds	$(-\infty, 1]$	$[3; 1]$
assignment	1	

Conflict: $\mathrm{res}_{x_4}(p_1, p_2) \rightsquigarrow p_4$ with *reason* $p_1 \wedge p_2$

UNSAT Example

$$\begin{array}{cccccccc} -y & + & x & & & \geq & 0 & (p_1) & p_1 \\ y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\ & & -x & + & 1 & \geq & 0 & (p_3) & p_3 \\ & & x & - & 3 & \geq & 0 & (p_4) & p_1 \wedge p_2 \end{array}$$

variable	x	
bounds		
assignment		

UNSAT Example

$$\begin{array}{rrrrrrl} -y & + & x & & & \geq & 0 & (p_1) & p_1 \\ y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\ & & -x & + & 1 & \geq & 0 & \color{blue}{(p_3)} & p_3 \\ & & x & - & 3 & \geq & 0 & \color{red}{(p_4)} & p_1 \wedge p_2 \end{array}$$

variable	x		
bounds	$[3; 1]$		
assignment			

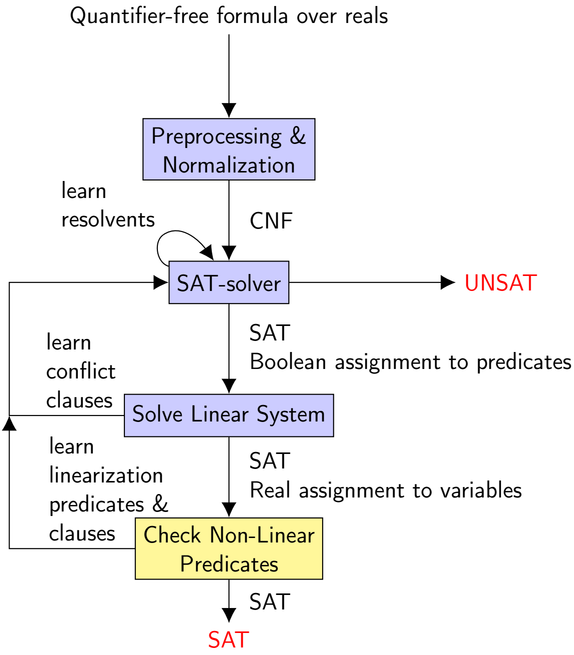
Conflict: $\mathrm{res}_{x_4}(p_3, p_4) \rightsquigarrow p_5$ with *reason* $p_3 \wedge p_1 \wedge p_2$

UNSAT Example

$$
\begin{array}{rrrrrcclc}
-y & + & x & & & \geq & 0 & (p_1) & p_1 \\
y & & & - & 3 & \geq & 0 & (p_2) & p_2 \\
 & & -x & + & 1 & \geq & 0 & (p_3) & p_3 \\
 & & x & - & 3 & \geq & 0 & (p_4) & p_1 \wedge p_2 \\
 & & & & -2 & \geq & 0 & (p_5) & p_3 \wedge p_1 \wedge p_2
\end{array}
$$

variable	x		
bounds	$[3; 1]$		
assignment			

Conflict: $\text{res}_{x_4}(p_3, p_4) \rightsquigarrow p_5$ with *reason* $p_3 \wedge p_1 \wedge p_2$ is constant $\bot$, UNSAT $\rightsquigarrow$ conflict clause is negation of reason of p_5


Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

Given a system of non-linear predicates and a real assignment $\alpha : \{x_1, \ldots, x_m\} \to \mathbb{R}_{\mathsf{alg}}$, for every non-linear predicate $P \risingdotseq \varphi \; \theta \; f(\psi)$ decide whether it is satisfied by α, if not generate an implied linearization

$$P \wedge D(\psi) \implies G(\psi, \varphi)$$

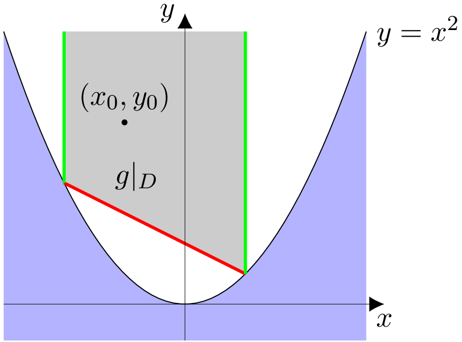
contradicting α. Otherwise, output SAT.

Note that D and G are linear predicates if φ and ψ are linear terms.

Compute (rational) values y_0 and x_0 of φ and ψ under α and decide algorithm based on known symbol $f \in \mathcal{F}$.

Example

Assume $P \leftrightharpoons y\ \theta\ x^2$, $\alpha(x) = x_0$, $\alpha(y) = y_0$ and $\neg(y_0\ \theta\ x_0^2)$ for some $\theta \in \{<, \leq, >, \geq\}$.



$D(x) \leftrightharpoons (x \in [a, b])$ and $g(x) = cx + d$ for rational a, b, c, d depending on (y_0, θ, x_0). Learn linearization $(P \wedge D(x) \implies G(x, y))$ where $G(x, y) \leftrightharpoons (y\ \theta\ g(x))$.

Classes of functions

If under α, $P \iff y_0\ \theta\ f(x_0)$ for $(x_0, y_0) \in \mathbb{Q}^2$ and $\neg(y_0\ \theta\ f(x_0))$.

Assumptions in order to compute linearization s.t. $P \wedge D \implies G$:

- $\theta \in \{<, \leq, >, \geq\}$, otherwise linear guessing generally won't work.
- θ decidable $\rightsquigarrow$ e.g. f maps rationals to rationals.

Classes of functions

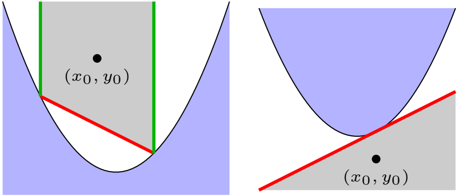
If under α, $P \iff y_0 \; \theta \; f(x_0)$ for $(x_0, y_0) \in \mathbb{Q}^2$ and $\neg(y_0 \; \theta \; f(x_0))$.

Assumptions in order to compute linearization s.t. $P \wedge D \implies G$:

- $\theta \in \{<, \leq, >, \geq\}$, otherwise linear guessing generally won't work.
- θ decidable $\rightsquigarrow$ e.g. f maps rationals to rationals.

Then have algorithms to exclude area $P \wedge D \wedge \neg G$ if

- f convex:



abs or $x \mapsto x^{2n}$ for $n \in \mathbb{N}$

Classes of functions

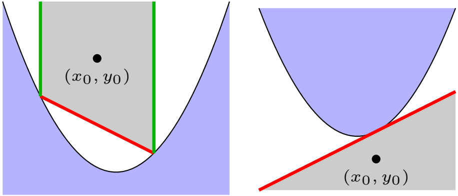
If under α, $P \iff y_0\ \theta\ f(x_0)$ for $(x_0, y_0) \in \mathbb{Q}^2$ and $\neg(y_0\ \theta\ f(x_0))$.

Assumptions in order to compute linearization s.t. $P \wedge D \implies G$:

- $\theta \in \{<, \leq, >, \geq\}$, otherwise linear guessing generally won't work.
- θ decidable $\rightsquigarrow$ e.g. f maps rationals to rationals.

Then have algorithms to exclude area $P \wedge D \wedge \neg G$ if

- f convex:



  abs or $x \mapsto x^{2n}$ for $n \in \mathbb{N}$

- f concave $\iff -f$ convex

Classes of functions

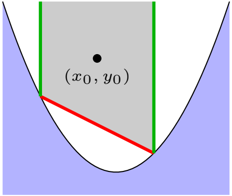
If under α, $P \iff y_0\ \theta\ f(x_0)$ for $(x_0, y_0) \in \mathbb{Q}^2$ and $\neg(y_0\ \theta\ f(x_0))$.

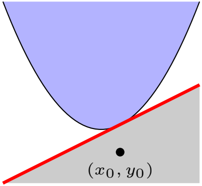
Assumptions in order to compute linearization s.t. $P \wedge D \implies G$:

- $\theta \in \{<, \leq, >, \geq\}$, otherwise linear guessing generally won't work.
- θ decidable $\rightsquigarrow$ e.g. f maps rationals to rationals.

Then have algorithms to exclude area $P \wedge D \wedge \neg G$ if

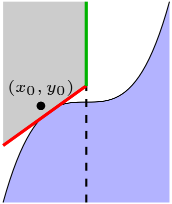
- f convex:





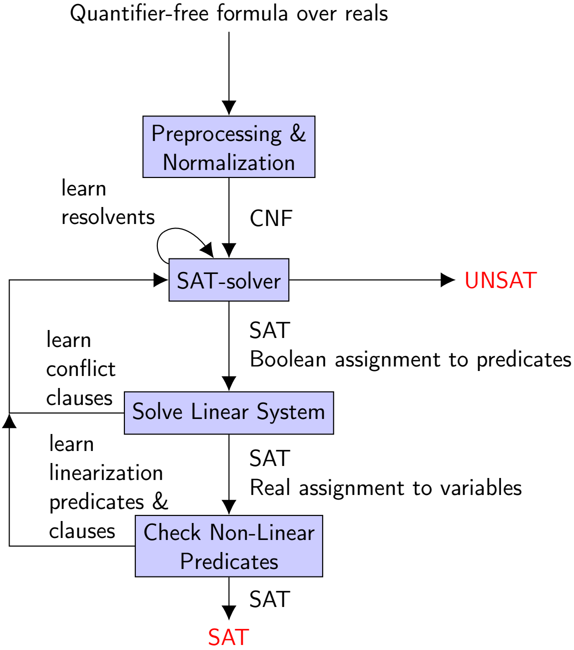
abs or $x \mapsto x^{2n}$ for $n \in \mathbb{N}$

- f piecewise convex/-cave:



$x \mapsto x^{2n+1}$ for $n \in \mathbb{N}$

- f concave $\iff -f$ convex

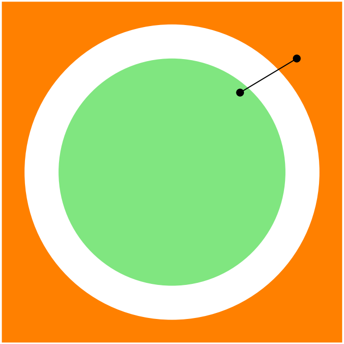

Quantifier-free formula over reals
Preprocessing & Normalization
CNF
learn resolvents
SAT-solver
UNSAT
SAT
Boolean assignment to predicates
learn conflict clauses
Solve Linear System
SAT
Real assignment to variables
learn linearization predicates & clauses
Check Non-Linear Predicates
SAT
SAT

Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \wedge \|\vec{y}\|_2 \geq \sqrt{64} \wedge \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.

Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \wedge \|\vec{y}\|_2 \geq \sqrt{64} \wedge \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.



Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \wedge \|\vec{y}\|_2 \geq \sqrt{64} \wedge \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.

s : 'sat'
u : 'unsat'
? : 'unknown'
>: timeout

r	ksmt		cvc4		z3		mathsat	
$\sqrt{37}$	u	0.03	u	0.76	u	510.67	u	40.55
$\sqrt{49}$	u	0.16	u	2.46	>	5h	u	6307.18
$\sqrt{62}$	u	7.15	u	5.07				
$\sqrt{63}$	u	27.43	?	0.48				
$\sqrt{64}$	s	0.01	?	0.01	s	0.00	>	21h

Timings on Linux, Core-i7 3.6GHz, (all except `mathsat`: g++-7.3.0)

Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \wedge \|\vec{y}\|_2 \geq \sqrt{64} \wedge \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.

s : 'sat'
u : 'unsat'
? : 'unknown'
>: timeout

r	ksmt		cvc4		z3		mathsat	
$\sqrt{37}$	u	0.03	u	0.76	u	510.67	u	40.55
$\sqrt{49}$	u	0.16	u	2.46	>	5h	u	6307.18
$\sqrt{62}$	u	7.15	u	5.07				
$\sqrt{63}$	u	27.43	?	0.48				
$\sqrt{64}$	s	0.01	?	0.01	s	0.00	>	21h

KK: $\exists \vec{x}_1, \ldots, \vec{x}_n \in \mathbb{R}^d : \bigwedge_{1 \leq i \leq n} \|\vec{x}_i\|_\infty \leq 1 \wedge \bigwedge_{1 \leq i < j \leq n} \|\vec{x}_i - \vec{x}_j\|_2 > 2$

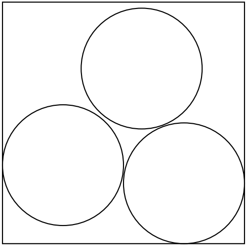
Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \land \|\vec{y}\|_2 \geq \sqrt{64} \land \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.

s : 'sat'
u : 'unsat'
? : 'unknown'
>: timeout

r	ksmt		cvc4		z3		mathsat	
$\sqrt{37}$	u	0.03	u	0.76	u	510.67	u	40.55
$\sqrt{49}$	u	0.16	u	2.46	>	5h	u	6307.18
$\sqrt{62}$	u	7.15	u	5.07				
$\sqrt{63}$	u	27.43	?	0.48				
$\sqrt{64}$	s	0.01	?	0.01	s	0.00	>	21h

KK: $\exists \vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d : \bigwedge_{1 \leq i \leq n} \|\vec{x}_i\|_\infty \leq 1 \land \bigwedge_{1 \leq i < j \leq n} \|\vec{x}_i - \vec{x}_j\|_2 > 2$



Benchmarks

BB: $\exists \vec{x}, \vec{y} \in \mathbb{R}^3 : \|\vec{x}\|_2 \leq r \wedge \|\vec{y}\|_2 \geq \sqrt{64} \wedge \|\vec{x} - \vec{y}\|_\infty \leq \frac{1}{100}$.

s : 'sat'
u : 'unsat'
? : 'unknown'
$>$: timeout

r	ksmt		cvc4		z3		mathsat	
$\sqrt{37}$	u	0.03	u	0.76	u	510.67	u	40.55
$\sqrt{49}$	u	0.16	u	2.46	$>$	5h	u	6307.18
$\sqrt{62}$	u	7.15	u	5.07				
$\sqrt{63}$	u	27.43	?	0.48				
$\sqrt{64}$	s	0.01	?	0.01	s	0.00	$>$	21h

KK: $\exists \vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d : \bigwedge_{1 \leq i \leq n} \|\vec{x}_i\|_\infty \leq 1 \wedge \bigwedge_{1 \leq i < j \leq n} \|\vec{x}_i - \vec{x}_j\|_2 > 2$

d	n	ksmt		cvc4		z3		mathsat		yices	
2	2	s	0.01	?	0.03	s	0.01	s	0.02	s	0.01
	3	s	0.03	?	0.08	$>$	60m	s	0.24	s	0.03
	4	$>$		u	1474.16	$>$	60m	u	8.11	$>$	17h
	5	u	1.43	u	0.45	$>$	8h	u	0.28	$>$	8h
	6	u	5.00	u	0.75	$>$	8h	u	0.40	$>$	166m
3	5	s	0.93	?	465.45	$>$	8h	s	0.12	s	0.06
	6	s	6.02	$>$	143m	$>$	7h	$>$	8h	$>$	6h
4	5	s	0.38	?	1544.87	s	2165.78	s	0.10	s	7.34
	6	s	0.57	$>$	91m	$>$	8h	s	0.23	s	0.38
	7	s	14.27	$>$	160m	$>$	8h	s	0.18	$>$	60m

Timings on Linux, Core-i7 3.6GHz, (all except `mathsat`: g++-7.3.0)

Future work:

- Explore class of linearizable functions.
  - exactly BSS?
  - what about $\sin p = q$ (p, q rational, real algebraic)?
- Merge solvers to find assignments incrementally.
  - Don't overload SAT-solver with linear resolvents.
- Borrow heuristics from SAT-solvers
  - e.g. for ordering real variables: VSIDS, ...
  - restarts, etc.
- Syntax transformation needs some love.