

On Gröbner bases in SMT-compliant decision procedures over the reals

Sebastian Junges, Ulrich Loup, Florian Corzilius, Erika Ábrahám

5th International Conference on Algebraic Informatics
September 5th, 2013



RWTHAACHEN
UNIVERSITY

Motivation

Are there solutions for x, y in the domain \mathbb{R} , such that:

$$y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Motivation

Are there solutions for x, y in the domain \mathbb{R} , such that:

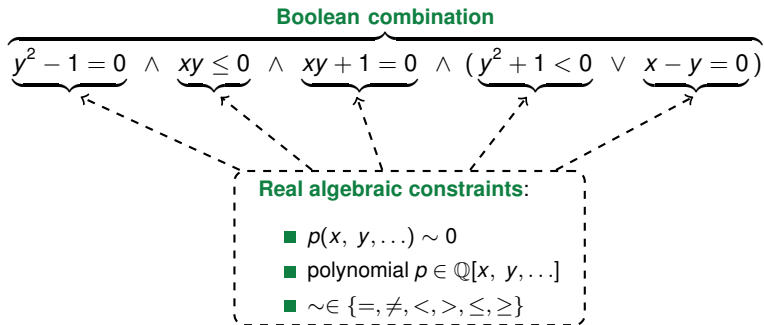
$$\underbrace{y^2 - 1 = 0} \wedge \underbrace{xy \leq 0} \wedge \underbrace{xy + 1 = 0} \wedge (\underbrace{y^2 + 1 < 0} \vee \underbrace{x - y = 0})$$

Real algebraic constraints:

- $p(x, y, \dots) \sim 0$
- polynomial $p \in \mathbb{Q}[x, y, \dots]$
- $\sim \in \{=, \neq, <, >, \leq, \geq\}$

Motivation

Are there solutions for x, y in the domain \mathbb{R} , such that:



Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y)$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1)$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

$$\text{red}_{\{x+1\}}(-2x + 2y)$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

$$\text{red}_{\{x+1\}}(-2x + 2y) = -2x + 2y - 2(x + 1) = -2y - 2$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

$$\text{red}_{\{x+1\}}(-2x + 2y) = -2x + 2y - 2(x + 1) = -2y - 2$$

$$\text{red}_{\{y+1\}}(-2y - 2)$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

$$\text{red}_{\{x+1\}}(-2x + 2y) = -2x + 2y - 2(x + 1) = -2y - 2$$

$$\text{red}_{\{y+1\}}(-2y - 2) = -2y - 2 - (-2)(y + 1) = 0$$

Reduction

$$p := 2 \cdot \overbrace{\underbrace{xy}_{\text{monomial}}}_{\text{term}} + 2y, \quad F := \{y + 1, x + 1\}$$

$$\text{red}_{\{y+1\}}(2xy + 2y) = 2xy + 2y - 2x(y + 1) = -2x + 2y$$

$$\text{red}_{\{x+1\}}(-2x + 2y) = -2x + 2y - 2(x + 1) = -2y - 2$$

$$\text{red}_{\{y+1\}}(-2y - 2) = -2y - 2 - (-2)(y + 1) = 0$$

| |
|-----------------------|
| $\text{red}_F(p) = 0$ |
|-----------------------|

Buchberger's algorithm

Input: set of polynomials P
Output: Gröbner basis G for $\langle P \rangle$

```
 $G := P$   
while true:  
   $G' := G$   
  for each  $\{p, q\} \subseteq G', p \neq q$ :  
     $s := \text{red}_G(S(p, q))$   
    if  $s \neq 0$ :  $G := U(G, s)$   
  if  $G = G'$ : break  
return  $G$ 
```

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

G := P
while true:
  G' := G
  for each  $\{p, q\} \subseteq G', p \neq q$ :
    s :=  $\text{red}_G(S(p, q))$ 
    if  $s \neq 0$ : G :=  $U(G, s)$ 
  if  $G = G'$ : break
return G
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

G := P
while true:
  G' := G
  for each  $\{p, q\} \subseteq G', p \neq q$ :
    s :=  $\text{red}_G(S(p, q))$ 
    if  $s \neq 0$ : G :=  $U(G, s)$ 
  if  $G = G'$ : break
return G
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

$$\begin{aligned}
 S(p, q) &= y \cdot p - x \cdot q \\
 &= xy^2 + y - xy^2 + x \\
 &= x + y
 \end{aligned}$$

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

G := P
while true:
  G' := G
  for each  $\{p, q\} \subseteq G', p \neq q$ :
    s :=  $\text{red}_G(S(p, q))$ 
    if  $s \neq 0$ : G :=  $U(G, s)$ 
  if  $G = G'$ : break
return G
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

$$\begin{aligned} S(p, q) &= y \cdot p - x \cdot q \\ &= xy^2 + y - xy^2 + x \\ &= x + y \end{aligned}$$

$$\text{red}_G(x + y) = x + y$$

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

 $G := P$ 
while true:
   $G' := G$ 
  for each  $\{p, q\} \subseteq G', p \neq q$ :
     $s := \text{red}_G(S(p, q))$ 
    if  $s \neq 0$ :  $G := U(G, s)$ 
  if  $G = G'$ : break
return  $G$ 
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

$$\begin{aligned} S(p, q) &= y \cdot p - x \cdot q \\ &= xy^2 + y - xy^2 + x \\ &= x + y \end{aligned}$$

$$\text{red}_G(x + y) = x + y$$

Here: $U(G, s) := G \cup \{s\}$

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

G := P
while true:
  G' := G
  for each  $\{p, q\} \subseteq G', p \neq q$ :
    s :=  $\text{red}_G(S(p, q))$ 
    if  $s \neq 0$ : G :=  $U(G, s)$ 
  if  $G = G'$ : break
return G
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

$$\begin{aligned} S(p, q) &= y \cdot p - x \cdot q \\ &= xy^2 + y - xy^2 + x \\ &= x + y \end{aligned}$$

$$\text{red}_G(x + y) = x + y$$

Here: $U(G, s) := G \cup \{s\}$

We want to use:

$$\text{reduced GB of } \langle P \rangle \text{ is } \{1\} \Leftrightarrow \overbrace{\{r \in \mathbb{C}^n \mid p(r) = 0 \text{ for all } p \in P\} = \emptyset}^{\text{no common zeros in } \mathbb{C}}$$

Buchberger's algorithm

Input: set of polynomials P
 Output: Gröbner basis G for $\langle P \rangle$

```

G := P
while true:
  G' := G
  for each  $\{p, q\} \subseteq G', p \neq q$ :
    s :=  $\text{red}_G(S(p, q))$ 
    if  $s \neq 0$ : G :=  $U(G, s)$ 
  if  $G = G'$ : break
return G
  
```

Example:

$$G := \{\overbrace{xy + 1}^p, \overbrace{y^2 - 1}^q\}$$

$$\begin{aligned} S(p, q) &= y \cdot p - x \cdot q \\ &= xy^2 + y - xy^2 + x \\ &= x + y \end{aligned}$$

$$\text{red}_G(x + y) = x + y$$

Here: $U(G, s) := G \cup \{s\}$

We want to use:

$$\begin{aligned} \text{reduced GB of } \langle P \rangle \text{ is } \{1\} &\Leftrightarrow \overbrace{\{r \in \mathbb{C}^n \mid p(r) = 0 \text{ for all } p \in P\} = \emptyset}^{\text{no common zeros in } \mathbb{C}} \\ &\Rightarrow \text{no common zeros in } \mathbb{R} \end{aligned}$$

Motivation

Are there solutions for x, y in the domain \mathbb{R} , such that:

$$y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

SAT solver

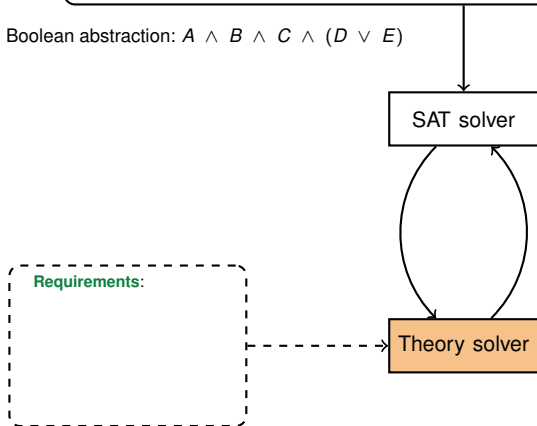
Theory solver

Requirements:

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$



SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$

true true true

SAT solver

add: $y^2 - 1 = 0, xy \leq 0, xy + 1 = 0$

Requirements:

Theory solver

$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0\}$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true

SAT solver

check consistency

answer: consistent

Requirements:

1 consistency check

Theory solver

$$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true true

SAT solver

add: $x - y = 0$

Requirements:

1 consistency check

Theory solver

$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0, x - y = 0\}$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true true

SAT solver

check consistency

answer: inconsistent

Requirements:

- 1 consistency check
- 2 incrementality

Theory solver

$$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0, x - y = 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true true

SAT solver

check consistency

answer: inconsistent

reason: $\{x - y = 0, xy + 1 = 0\}$

Requirements:

- 1 consistency check
- 2 incrementality
- 3 infeasible subsets

Theory solver

$$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0, x - y = 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true false

SAT solver

delete: $x - y = 0$

Requirements:

- 1 consistency check
- 2 incrementality
- 3 infeasible subsets
- 4 backtracking

Theory solver

$$\{y^2 - 1 = 0, xy \leq 0, xy + 1 = 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true true false

SAT solver

add: $y^2 + 1 < 0$

Requirements:

- 1 consistency check
- 2 incrementality
- 3 infeasible subsets
- 4 backtracking

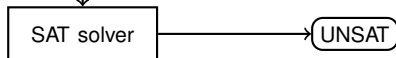
Theory solver

$$\{x - y = 0, xy \leq 0, y^2 + 1 < 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

Boolean abstraction: $A \wedge B \wedge C \wedge (D \vee E)$
 true true true true false



check consistency

answer: inconsistent
 reason: $\{y^2 + 1 < 0\}$

Requirements:

- 1 consistency check
- 2 incrementality
- 3 infeasible subsets
- 4 backtracking

Theory solver

$$\{x - y = 0, xy \leq 0, y^2 + 1 < 0\}$$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$

SAT solver

Theory solver

Requirements:

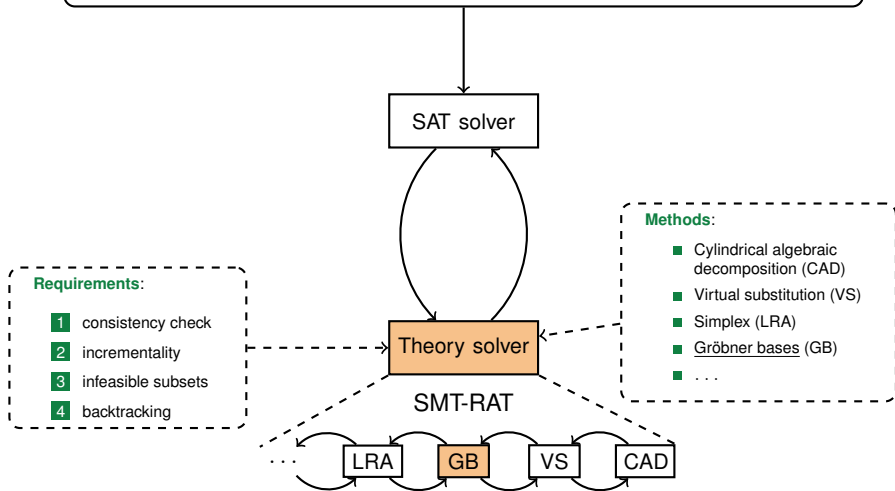
- 1 consistency check
- 2 incrementality
- 3 infeasible subsets
- 4 backtracking

Methods:

- Cylindrical algebraic decomposition (CAD)
- Virtual substitution (VS)
- Simplex (LRA)
- Gröbner bases (GB)
- ...

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$



The SMT-compliant Gröbner module

1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

1.) add: $p_1 = 0$, $p_2 \leq 0$, $p_3 = 0$

Polynomials of equations:

$$\{p_1, p_3\}$$

Received equations:

$$\{p_1 = 0, p_3 = 0\}$$

Received constraints:

$$\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$$

The SMT-compliant Gröbner module

1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, \emptyset)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, \emptyset)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency

GB of $\{p_1, p_3\}$ is $G_1 \neq \{1\}$

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, \emptyset)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency

GB of $\{p_1, p_3\}$ is $G_1 \neq \{1\}$

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency

GB of $\{p_1, p_3\}$ is $G_1 \neq \{1\}$

\Rightarrow do not know!

Solver state stack:

$(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations:

$\{p_1, p_3\}$

Received equations:

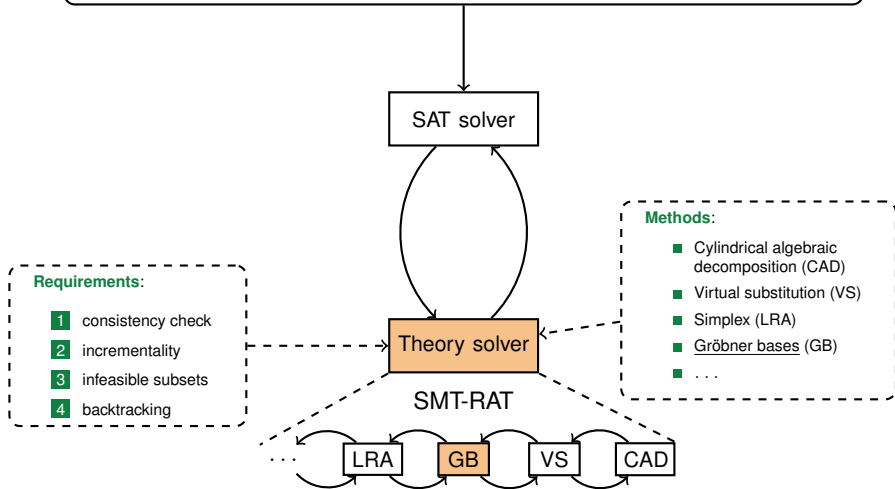
$\{p_1 = 0, p_3 = 0\}$

Received constraints:

$\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

SMT solver

$$\exists x, y : y^2 - 1 = 0 \wedge xy \leq 0 \wedge xy + 1 = 0 \wedge (y^2 + 1 < 0 \vee x - y = 0)$$



The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency

GB of $\{p_1, p_3\}$ is $G_1 \neq \{1\}$ \Rightarrow do not know!
 ask backend: $\{g = 0 \mid g \in G_1\} \cup \{p_2 \leq 0\}$ \Rightarrow consistent!

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$

Solver state stack: $(\{p_5 = 0\}, \emptyset)$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) **check consistency**

Solver state stack: $(\{p_5 = 0\}, \emptyset)$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency

GB of $G_1 \cup \{p_5\}$ is $\{1\}$

Solver state stack: $(\{p_5 = 0\}, \{1\})$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency

GB of $G_1 \cup \{p_5\}$ is $\{1\}$ \Rightarrow inconsistent!

Solver state stack: $(\{p_5 = 0\}, \{1\})$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency

GB of $G_1 \cup \{p_5\}$ is $\{1\}$
 reason: received equations

\Rightarrow inconsistent!

Solver state stack:

$$(\{p_5 = 0\}, \{1\})$$

$$(\{p_1 = 0, p_3 = 0\}, G_1)$$

Polynomials of equations:

$$\{p_1, p_3, p_5\}$$

Received equations:

$$\{p_1 = 0, p_3 = 0, p_5 = 0\}$$

Received constraints:

$$\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency

GB of $G_1 \cup \{p_5\}$ is $\{1\}$
 reason: received equations
 better: $\{p = 0 \mid p \in C_{org}(1)\}$
 $C_{org}(s) := C_{org}(p) \cup C_{org}(q)$

\Rightarrow inconsistent!

with
 if $s := red_G(S(p, q))$

$(\{p_5 = 0\}, \{1\})$

Solver state stack:

$(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations:

$\{p_1, p_3, p_5\}$

Received equations:

$\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints:

$\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency

Solver state stack: $(\{p_5 = 0\}, \{1\})$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency
- 5.) remove: $p_5 = 0$

Solver state stack: $(\{p_5 = 0\}, \{1\})$
 $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3, p_5\}$

Received equations: $\{p_1 = 0, p_3 = 0, p_5 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0, p_5 = 0\}$

The SMT-compliant Gröbner module

- 1.) add: $p_1 = 0, p_2 \leq 0, p_3 = 0$
- 2.) check consistency
- 3.) add: $p_5 = 0$
- 4.) check consistency
- 5.) remove: $p_5 = 0$
- 6.) ...

Solver state stack: $(\{p_1 = 0, p_3 = 0\}, G_1)$

Polynomials of equations: $\{p_1, p_3\}$

Received equations: $\{p_1 = 0, p_3 = 0\}$

Received constraints: $\{p_1 = 0, p_2 \leq 0, p_3 = 0\}$

Handling inequalities

Approach 1: (transform inequalities to equations)

- Squares are exactly the non-negative numbers
- Only the zero has no multiplicative inverse

Handling inequalities

Approach 1: (transform inequalities to equations)

- Squares are exactly the non-negative numbers
- Only the zero has no multiplicative inverse
- Example:

$$p > 0 \iff \exists y. py^2 - 1 = 0$$

Handling inequalities

Approach 1: (transform inequalities to equations)

- Squares are exactly the non-negative numbers
- Only the zero has no multiplicative inverse
- Example:

$$p > 0 \iff \exists y. py^2 - 1 = 0$$

Approach 2: (reduce inequalities)

- Reduce polynomial in inequality $p \sim 0$ with current GB G : $red_G(p) = q$

Handling inequalities

Approach 1: (transform inequalities to equations)

- Squares are exactly the non-negative numbers
- Only the zero has no multiplicative inverse
- Example:

$$p > 0 \iff \exists y. py^2 - 1 = 0$$

Approach 2: (reduce inequalities)

- Reduce polynomial in inequality $p \sim 0$ with current GB G : $red_G(p) = q$
- If $q \in \mathbb{Q}$ (constant) and ...
 - 1 ... $q \sim 0$: do not pass $p \sim 0$ to a backend
 - 2 ... otherwise: conflict with reason $\{p = 0 \mid p \in C_{org}(q)\} \cup \{p \sim 0\}$

Further optimizations

Assume, G is the current GB:

Further optimizations

Assume, G is the current GB:

- Idea: apart from reasons pass further **theory tautologies** to the SAT solver

Further optimizations

Assume, G is the current GB:

- Idea: apart from reasons pass further **theory tautologies** to the SAT solver
 - If we receive a constraint $p \sim 0$ and $red_G(p) = q \in \mathbb{Q}$ learn:

$$\left(\bigwedge_{s \in C_{org}(q)} s = 0 \right) \rightarrow (p = 0)$$

- Reduces Boolean complexity for SAT solving

Further optimizations

Assume, G is the current GB:

- Idea: apart from reasons pass further **theory tautologies** to the SAT solver

- If we receive a constraint $p \sim 0$ and $red_G(p) = q \in \mathbb{Q}$ learn:

$$\left(\bigwedge_{s \in C_{org}(q)} s = 0 \right) \rightarrow (p = 0)$$

- Reduces Boolean complexity for SAT solving

- Idea: $t - x$ in GB G with $x \notin t \Rightarrow$ update $G = (G \setminus \{t - x\})[t \setminus x]$
(**iterative variable elimination**)

Further optimizations

Assume, G is the current GB:

- Idea: apart from reasons pass further **theory tautologies** to the SAT solver

- If we receive a constraint $p \sim 0$ and $red_G(p) = q \in \mathbb{Q}$ learn:

$$\left(\bigwedge_{s \in C_{org}(q)} s = 0 \right) \rightarrow (p = 0)$$

- Reduces Boolean complexity for SAT solving
- Idea: $t - x$ in GB G with $x \notin t \Rightarrow$ update $G = (G \setminus \{t - x\})[t \setminus x]$
(**iterative variable elimination**)
 - Apply it repeatedly until a fixpoint is reached
 - Take also the inequalities into account

Experimental results

Made on a 2.1 GHz AMD Xeon core machine with a time-out of 200 seconds and a memory limit of 4 GB (per instance).

| | BOUNCE (180) | | MET (8276) | | KEY (421) | | |
|---|--------------|-------|------------|---------|-----------|-------|-------|
| | # | time | # | time | # | time | |
| S _{ref} | 101 | 135.7 | 7107 | 16350.4 | 385 | 573.6 | } ... |
| - sat | 84 | 135.3 | 4780 | 8507.4 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2327 | 7843.0 | 385 | 573.6 | |
| GB | 101 | 135.7 | 7088 | 16149.7 | 408 | 381.9 | } ... |
| - sat | 84 | 135.1 | 4774 | 7985.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2314 | 8164.2 | 408 | 381.9 | |
| GB ^{ItVarElim} | 101 | 138.4 | 7096 | 16167.4 | 407 | 376.1 | } ... |
| - sat | 84 | 137.8 | 4773 | 8077.0 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2323 | 8090.5 | 407 | 376.1 | |
| GB ^{passGBEQ} | 93 | 73.7 | 7037 | 16484.2 | 405 | 68.6 | } ... |
| - sat | 76 | 73.3 | 4776 | 8195.6 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2261 | 8288.7 | 405 | 68.6 | |
| GB ^{ItVarElim} _{passGBEQ} | 127 | 32.7 | 7037 | 15794.5 | 405 | 72.2 | } ... |
| - sat | 81 | 18.3 | 4777 | 7976.3 | 0 | 0.0 | |
| - unsat | 46 | 14.4 | 2260 | 7818.2 | 405 | 72.2 | |
| GB ^{transEQ} | 101 | 142.9 | 6850 | 17840.2 | 409 | 385.3 | } ... |
| - sat | 84 | 142.3 | 4671 | 9633.1 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2179 | 8207.0 | 409 | 385.3 | |
| GB ^{ItVarElim} _{transEQ} | 101 | 136.0 | 6875 | 17767.9 | 412 | 377.8 | } ... |
| - sat | 84 | 135.4 | 4666 | 9768.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2209 | 7999.3 | 412 | 377.8 | |



Experimental results

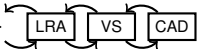






Made on a 2.1 GHz AMD Xeon core machine with a time-out of 200 seconds and a memory limit of 4 GB (per instance).

| | BOUNCE (180) | | MET (8276) | | KEY (421) | | |
|-----------------------------|--------------|-------|------------|---------|-----------|-------|-------|
| | # | time | # | time | # | time | |
| S_{ref} | 101 | 135.7 | 7107 | 16350.4 | 385 | 573.6 | } ... |
| - sat | 84 | 135.3 | 4780 | 8507.4 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2327 | 7843.0 | 385 | 573.6 | |
| GB | 101 | 135.7 | 7088 | 16149.7 | 408 | 381.9 | } ... |
| - sat | 84 | 135.1 | 4774 | 7985.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2314 | 8164.2 | 408 | 381.9 | |
| $GB^{ItVarElim}$ | 101 | 138.4 | 7096 | 16167.4 | 407 | 376.1 | } ... |
| - sat | 84 | 137.8 | 4773 | 8077.0 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2323 | 8090.5 | 407 | 376.1 | |
| $GB^{passGBEQ}$ | 93 | 73.7 | 7037 | 16484.2 | 405 | 68.6 | } ... |
| - sat | 76 | 73.3 | 4776 | 8195.6 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2261 | 8288.7 | 405 | 68.6 | |
| $GB^{ItVarElim}_{passGBEQ}$ | 127 | 32.7 | 7037 | 15794.5 | 405 | 72.2 | } ... |
| - sat | 81 | 18.3 | 4777 | 7976.3 | 0 | 0.0 | |
| - unsat | 46 | 14.4 | 2260 | 7818.2 | 405 | 72.2 | |
| $GB^{transEQ}$ | 101 | 142.9 | 6850 | 17840.2 | 409 | 385.3 | } ... |
| - sat | 84 | 142.3 | 4671 | 9633.1 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2179 | 8207.0 | 409 | 385.3 | |
| $GB^{ItVarElim}_{transEQ}$ | 101 | 136.0 | 6875 | 17767.9 | 412 | 377.8 | } ... |
| - sat | 84 | 135.4 | 4666 | 9768.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2209 | 7999.3 | 412 | 377.8 | |



Experimental results

Made on a 2.1 GHz AMD Xeon core machine with a time-out of 200 seconds and a memory limit of 4 GB (per instance).

| | BOUNCE (180) | | MET (8276) | | KEY (421) | | |
|----------------------------|--------------|-------|-------------|---------|-----------|-------|--|
| | # | time | # | time | # | time | |
| S_{ref} | 101 | 135.7 | 7107 | 16350.4 | 385 | 573.6 | } ...  |
| - sat | 84 | 135.3 | 4780 | 8507.4 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2327 | 7843.0 | 385 | 573.6 | |
| GB | 101 | 135.7 | 7088 | 16149.7 | 408 | 381.9 | } ...  |
| - sat | 84 | 135.1 | 4774 | 7985.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2314 | 8164.2 | 408 | 381.9 | |
| $GB^{ItVarElim}$ | 101 | 138.4 | 7096 | 16167.4 | 407 | 376.1 | } ...  |
| - sat | 84 | 137.8 | 4773 | 8077.0 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2323 | 8090.5 | 407 | 376.1 | |
| $GB^{passGBEQ}$ | 93 | 73.7 | 7037 | 16484.2 | 405 | 68.6 | } ...  |
| - sat | 76 | 73.3 | 4776 | 8195.6 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2261 | 8288.7 | 405 | 68.6 | |
| $GB^{ItVarElim\ passGBEQ}$ | 127 | 32.7 | 7037 | 15794.5 | 405 | 72.2 | } ...  |
| - sat | 81 | 18.3 | 4777 | 7976.3 | 0 | 0.0 | |
| - unsat | 46 | 14.4 | 2260 | 7818.2 | 405 | 72.2 | |
| $GB^{transEQ}$ | 101 | 142.9 | 6850 | 17840.2 | 409 | 385.3 | } ...  |
| - sat | 84 | 142.3 | 4671 | 9633.1 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2179 | 8207.0 | 409 | 385.3 | |
| $GB^{ItVarElim\ transEQ}$ | 101 | 136.0 | 6875 | 17767.9 | 412 | 377.8 | } ...  |
| - sat | 84 | 135.4 | 4666 | 9768.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2209 | 7999.3 | 412 | 377.8 | |

Experimental results

Made on a 2.1 GHz AMD Xeon core machine with a time-out of 200 seconds and a memory limit of 4 GB (per instance).

| | BOUNCE (180) | | MET (8276) | | KEY (421) | | |
|-------------------------------------|--------------|-------|------------|---------|-----------|-------|-------|
| | # | time | # | time | # | time | |
| S _{ref} | 101 | 135.7 | 7107 | 16350.4 | 385 | 573.6 | } ... |
| - sat | 84 | 135.3 | 4780 | 8507.4 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2327 | 7843.0 | 385 | 573.6 | |
| GB | 101 | 135.7 | 7088 | 16149.7 | 408 | 381.9 | } ... |
| - sat | 84 | 135.1 | 4774 | 7985.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2314 | 8164.2 | 408 | 381.9 | |
| GB ^{ItVarElim} | 101 | 138.4 | 7096 | 16167.4 | 407 | 376.1 | } ... |
| - sat | 84 | 137.8 | 4773 | 8077.0 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2323 | 8090.5 | 407 | 376.1 | |
| GB ^{passGBEQ} | 93 | 73.7 | 7037 | 16484.2 | 405 | 68.6 | } ... |
| - sat | 76 | 73.3 | 4776 | 8195.6 | 0 | 0.0 | |
| - unsat | 17 | 0.5 | 2261 | 8288.7 | 405 | 68.6 | |
| GB ^{ItVarElim} passGBEQ | 127 | 32.7 | 7037 | 15794.5 | 405 | 72.2 | } ... |
| - sat | 81 | 18.3 | 4777 | 7976.3 | 0 | 0.0 | |
| - unsat | 46 | 14.4 | 2260 | 7818.2 | 405 | 72.2 | |
| GB ^{transEQ} | 101 | 142.9 | 6850 | 17840.2 | 409 | 385.3 | } ... |
| - sat | 84 | 142.3 | 4671 | 9633.1 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2179 | 8207.0 | 409 | 385.3 | |
| GB ^{ItVarElim} transEQ | 101 | 136.0 | 6875 | 17767.9 | 412 | 377.8 | } ... |
| - sat | 84 | 135.4 | 4666 | 9768.6 | 0 | 0.0 | |
| - unsat | 17 | 0.6 | 2209 | 7999.3 | 412 | 377.8 | |



Outlook

- Implementing further modules based on Gröbner bases:
 - Quantifier elimination
 - Computing realizable sign conditions [CAI'11]
- Optimize heuristics (variable ordering)
- Optimize real radical preserving update operator