CVC4 1.5 for Sygus Comp 2015

- CVC4 is an SMT solver
  - Fourth generation of Cooperating Validity Checker (CVC, CVC Lite, CVC3, CVC4)
  - Supports many ground theories:
    - Linear arithmetic, bitvectors, UF, datatypes, arrays, sets, strings, …
  - Supports quantified formulas
  - Two new approaches for refutation-based synthesis **[CAV 15]**
    1. Single-invocation properties
    2. Syntax-guided synthesis (SyGuS) problems
- Submission for Sygus Comp 2015 was joint work between:
  - EPFL: Andrew Reynolds, Viktor Kuncak
  - University of Iowa: Cesare Tinelli
  - NYU: Clark Barrett, Morgan Deters
  - Verimag: Tim King

Refutation-Based Synthesis

$$\exists\, \texttt{f}.\ \forall\, \texttt{xy}.\ (\texttt{f(x,y)} \geq \texttt{x} \wedge \texttt{f(x,y)} \geq \texttt{y} \wedge (\texttt{f(x,y)=x} \vee \texttt{f(x,y)=y}))$$

- Example: find a function `f` that computes max of two integers

Refutation-Based Synthesis

$$\exists f.\ \forall xy.\ \text{isMax}(f(x,y),x,y)$$

Refutation-Based Synthesis

$\exists$ `f.` $\forall$ `xy.isMax(f(x,y),x,y)`

Find model for `f` that satisfies this property

Refutation-Based Synthesis

$\exists f. \forall xy. isMax(f(x,y),x,y)$

Negate

$\forall f. \exists xy. \neg isMax(f(x,y),x,y)$

Instead, show negated formula is *unsatisfiable*

Refutation-Based Synthesis

$$\exists\, \mathtt{f}.\ \forall\, \mathtt{xy}.\mathtt{isMax}(\mathtt{f}(\mathtt{x},\mathtt{y}),\mathtt{x},\mathtt{y})$$

Negate

$$\boldsymbol{\forall \mathtt{f}}.\ \exists\, \mathtt{xy}.\neg\mathtt{isMax}(\mathtt{f}(\mathtt{x},\mathtt{y}),\mathtt{x},\mathtt{y})$$

- Eliminate second-order quantification over $\mathtt{f}$ in two ways

Refutation-Based Synthesis

$\exists f.\ \forall xy.\mathtt{isMax}(f(x,y),x,y)$

↓ Negate

$\forall f.\ \exists xy.\neg\mathtt{isMax}(\mathbf{f(x,y)},x,y)$

If single invocation, replace `f` with (first-order) variable `g`

$\exists xy.\mathbf{\forall g}.\neg\mathtt{isMax}(\mathbf{g},x,y)$

$\Rightarrow$ `g` represents the return value of `f`

Refutation-Based Synthesis

$\exists f. \forall xy.\text{isMax}(f(x,y),x,y)$

Negate

$\forall f. \exists xy.\neg\text{isMax}(\mathbf{f(x,y)},x,y)$

If single invocation, replace `f` with (first-order) variable `g`

$\exists xy.\forall g.\neg\text{isMax}(g,x,y)$

Otherwise, replace `f` with datatype `d`, and operator `ev`

```
D := zero | one | plus( D1, D2 ) | …
∀d.∃xy.¬isMax(ev(d,x,y),x,y)
∀dxy.ev(d,x,y)=…
```

⇒ `D` models the domain of possible solutions for `f`

Refutation-Based Synthesis

$\exists f. \forall xy.\mathtt{isMax}(f(x,y),x,y)$

Negate

$\forall f. \exists xy.\neg\mathtt{isMax}(f(x,y),x,y)$

If single invocation, replace `f` with (first-order) variable `g`

$\exists xy.\forall g.\neg\mathtt{isMax}(g,x,y)$

Single invocation approach

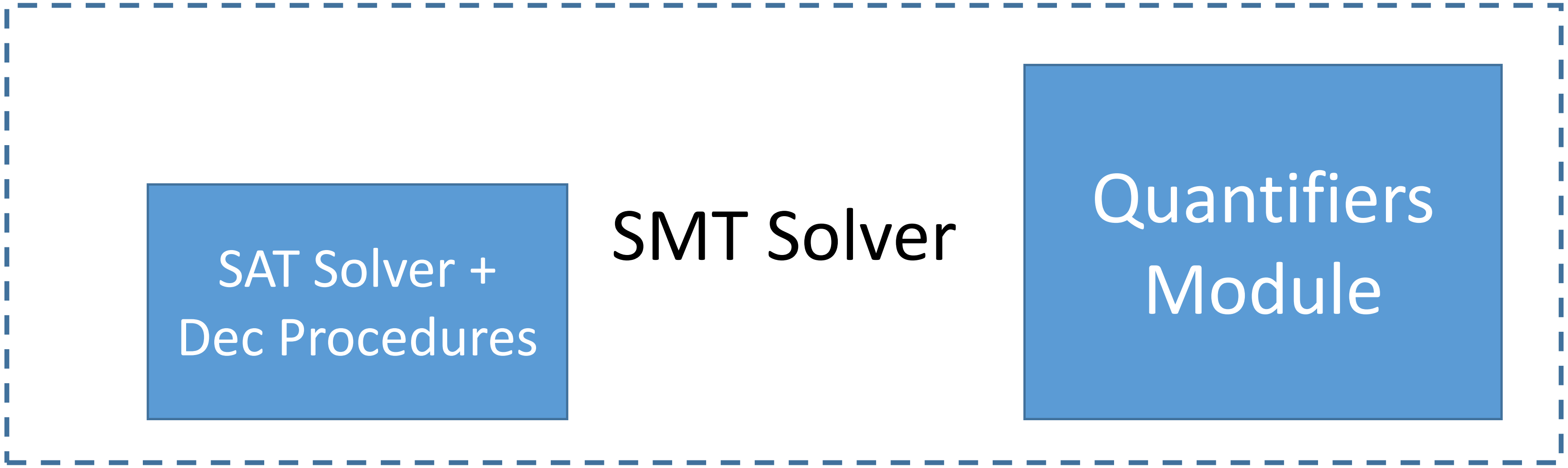
Otherwise, replace `f` with datatype `d`, and operator `ev`

```
D := zero | one | plus( D1, D2 ) | …
∀d.∃xy.¬isMax(ev(d,x,y),x,y)
∀dxy.ev(d,x,y)=…
```

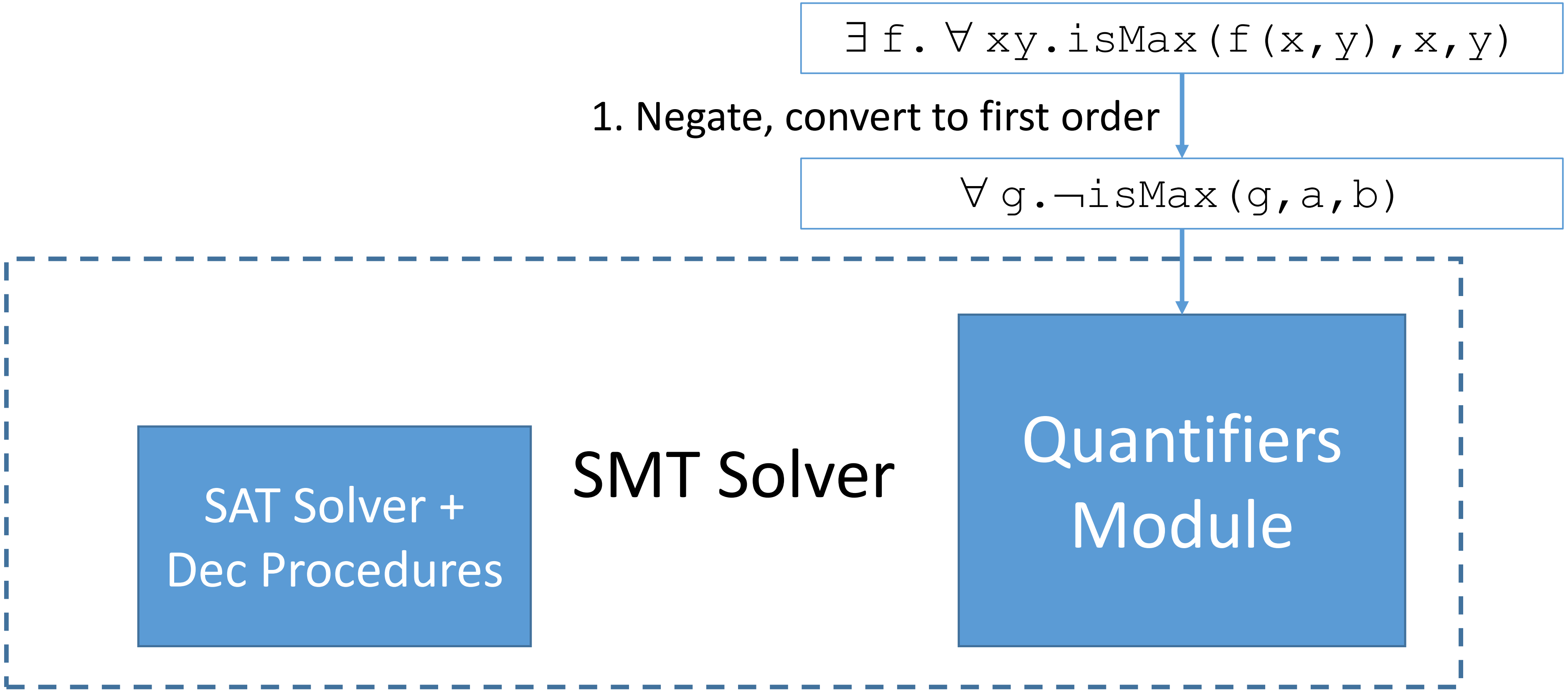
Syntax-guided approach

Solving Synthesis Conjectures in an SMT Solver

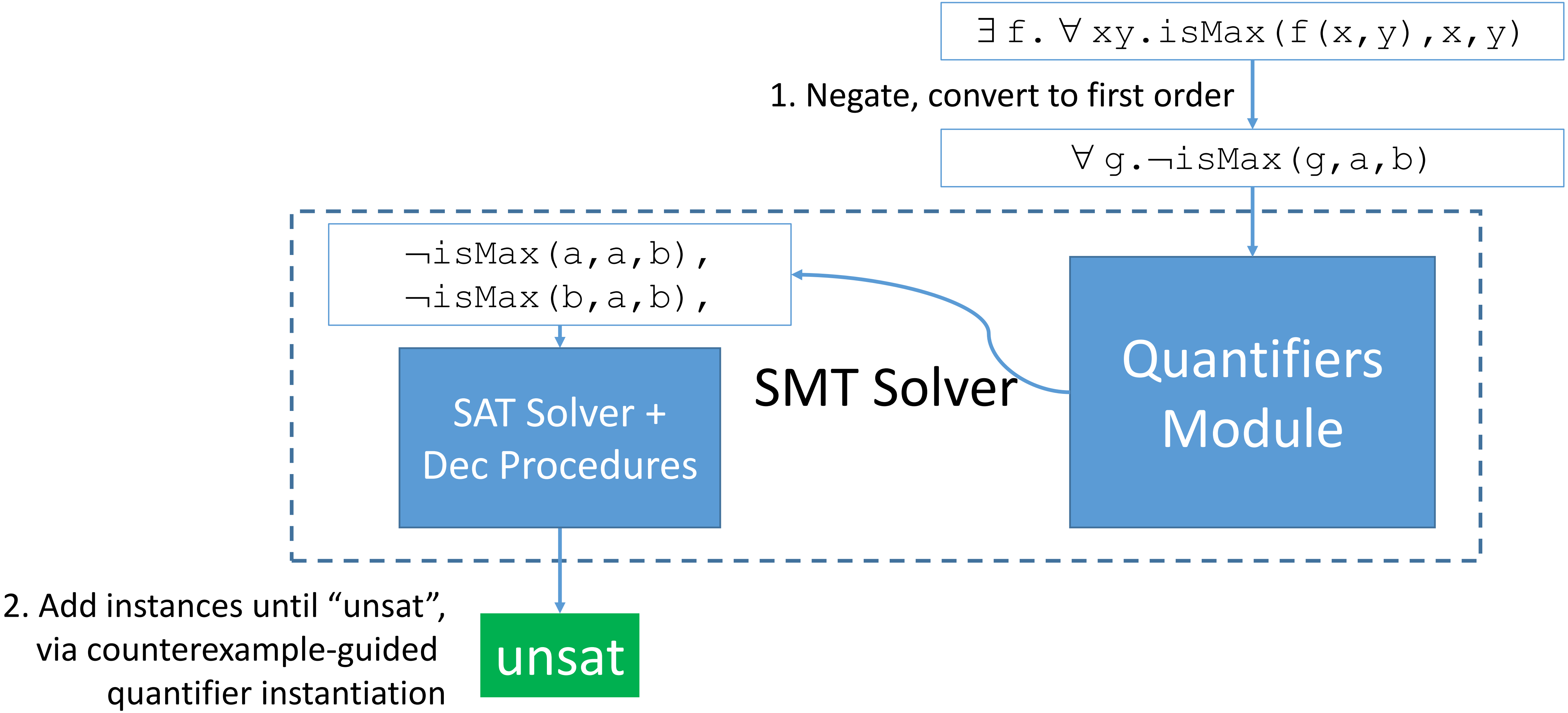
$\exists$ `f.` $\forall$ `xy.isMax(f(x,y),x,y)`



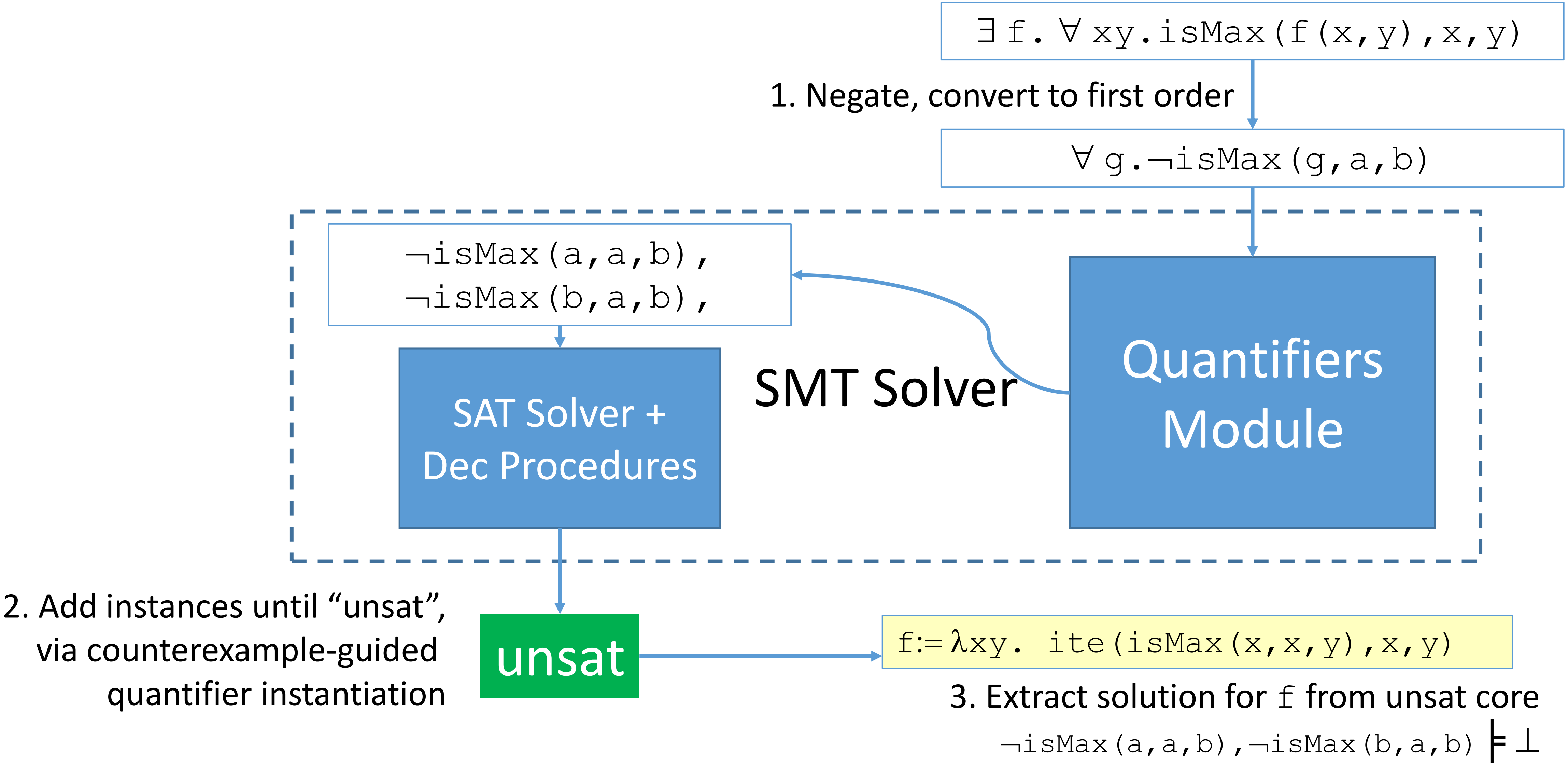
Solving Synthesis Conjectures in an SMT Solver



Solving Synthesis Conjectures in an SMT Solver



Solving Synthesis Conjectures in an SMT Solver



CVC4 in Sygus Comp 2015

- Entered all three tracks (General, LIA, INV)
  - For general/LIA track:
    - Most benchmarks are single invocation
    - Solution reconstruction methods to match syntactic restrictions, if necessary
  - For INV track:
    - All benchmarks are not single invocation
      - Due to form of benchmarks, for transition relations T:

$$\exists \, inv. \, \forall \, x. \, (inv(\mathbf{x}) \wedge T(x, x')) \Rightarrow inv(\mathbf{x'})$$

$\Rightarrow$ Resorts to syntax-guided approach