Specification Synthesis with Constrained Horn Clauses

PLDI'21 Distinguished Paper

Sumanth Prabhu S
TCS Research, IISc

FM Update Meeting
09 July 2021

Specification Synthesis with Constrained Horn Clauses

Sumanth Prabhu S

TCS Research, IISc, India

sumanth.prabhu@tcs.com

Grigory Fedyukovich

Florida State University, USA

grigory@cs.fsu.edu

Kumar Madhukar

TCS Research, India

kumar.madhukar@tcs.com

Deepak D'Souza

IISc, India

deepakd@iisc.ac.in

Specification Synthesis with Constrained Horn Clauses

Sumanth Prabhu S

TCS Research, IISc, India
sumanth.prabhu@tcs.com

Grigory Fedyukovich

Florida State University, USA
grigory@cs.fsu.edu

Kumar Madhukar

TCS Research, India
kumar.madhukar@tcs.com

Deepak D'Souza

IISc, India
deepakd@iisc.ac.in

Specification Synthesis with Constrained Horn Clauses

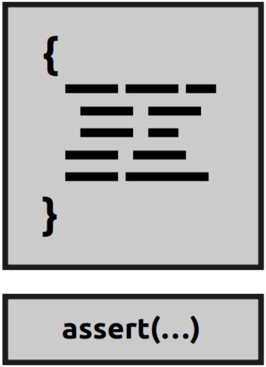
Sumanth Prabhu S
TCS Research, IISc, India
sumanth.prabhu@tcs.com

Grigory Fedyukovich
Florida State University, USA
grigory@cs.fsu.edu

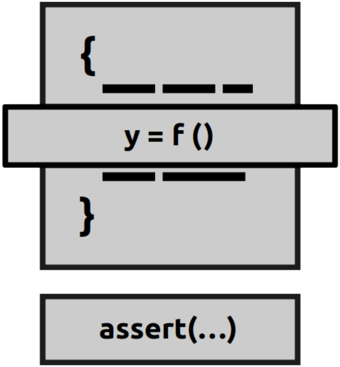
Kumar Madhukar
TCS Research, India
kumar.madhukar@tcs.com

Deepak D'Souza
IISc, India
deepakd@iisc.ac.in

Open Program Verification



Open Program Verification



Specification Synthesis

- Open Program Verification
- Compositional Verification
- Safety Games and many more

reduces to

Given a set of conditions with unknown functions
find quality specifications for the functions

Running Example

```
int x = 19;
```

$$\forall x\,.\, x = 19 \implies \mathrm{inv}(x)$$

```
while (*) {
  x = x - 1;
}
```

$$\forall x, x'\,.\, \mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$$

```
int y = f();
assert (x ≤ y);
```

$$\forall x, y\,.\, \mathrm{inv}(x) \wedge \mathrm{f}(y) \implies x \leq y$$

Program

Constrained Horn Clauses

CHCs

Constrained Horn Clauses

$$\forall \vec{x} \,.\, \varphi(\vec{x}) \implies \mathsf{r}(\vec{x}) \qquad \text{(fact)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} \mathsf{r}_i(\vec{x}_i) \wedge \psi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \mathsf{r}_{n+1}(\vec{x}_{n+1}) \qquad \text{(inductive)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} \mathsf{r}_i(\vec{x}_i) \wedge \pi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \bot \qquad \text{(query)}$$

CHCs

Constrained Horn Clauses

$$\forall \vec{x} \,.\, \varphi(\vec{x}) \implies r(\vec{x}) \qquad \text{(fact)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} r_i(\vec{x}_i) \wedge \psi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies r_{n+1}(\vec{x}_{n+1}) \qquad \text{(inductive)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} r_i(\vec{x}_i) \wedge \pi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \bot \qquad \text{(query)}$$

CHCs

Constrained Horn Clauses

$$\forall \vec{x} \,.\, \varphi(\vec{x}) \implies \mathrm{r}(\vec{x}) \qquad \text{(fact)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} \mathrm{r}_i(\vec{x}_i) \wedge \psi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \mathrm{r}_{n+1}(\vec{x}_{n+1}) \qquad \text{(inductive)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \le i \le n} \mathrm{r}_i(\vec{x}_i) \wedge \pi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \bot \qquad \text{(query)}$$

CHCs

Constrained Horn Clauses

$$\forall \vec{x} \,.\, \varphi(\vec{x}) \implies \mathrm{r}(\vec{x}) \qquad \text{(fact)}$$

$$\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \leq i \leq n} \mathrm{r}_i(\vec{x}_i) \wedge \psi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \mathrm{r}_{n+1}(\vec{x}_{n+1}) \qquad \text{(inductive)}$$

$$\boxed{\forall \vec{x}_1 \ldots \vec{x}_{n+1} \,.\, \bigwedge_{1 \leq i \leq n} \mathrm{r}_i(\vec{x}_i) \wedge \pi(\vec{x}_1, \ldots, \vec{x}_{n+1}) \implies \bot} \qquad \text{(query)}$$

CHCs - Example

$$\forall x\,.\,x = 19 \implies \mathrm{inv}(x)$$

$$\forall x, x'\,.\,\mathrm{inv}(x) \land x' = x - 1 \implies \mathrm{inv}(x')$$

$$\forall x, y\,.\,\mathrm{inv}(x) \land \mathrm{f}(y) \land \neg(x \leq y) \implies \bot$$

Solution to CHCs

- Given

  S a set of CHCs
  over relations $R = \{r_1 \ldots r_{n+1}\}$

- Find

  $M : R \rightarrow Predicates$
  M makes each CHC in S *valid*

Solution Quality

$$x = 19 \implies \text{inv}(x)$$

$$\text{inv}(x) \wedge x' = x - 1 \implies \text{inv}(x')$$

$$\text{inv}(x) \wedge \text{f}(y) \implies x \leq y$$

inv(x)	f(y)	
$x \leq 19$	*false*	**Vacuous**

Solution Quality

$$x = 19 \implies \mathsf{inv}(x)$$

$$\mathsf{inv}(x) \wedge x' = x - 1 \implies \mathsf{inv}(x')$$

$$\mathsf{inv}(x) \wedge \mathsf{f}(y) \implies x \leq y$$

$\mathsf{inv}(x)$	$\mathsf{f}(y)$	
$x \leq 19$	*false*	**Vacuous**
$x \leq 19$	$y = 19$	**Non-Vacuous but Non-Maximal**

Solution Quality

$$x = 19 \implies \mathrm{inv}(x)$$

$$\mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$$

$$\mathrm{inv}(x) \wedge \mathrm{f}(y) \implies x \leq y$$

inv(x)	f(y)	
$x \leq 19$	*false*	**Vacuous**
$x \leq 19$	$y = 19$	**Non-Vacuous but Non-Maximal**
$x \leq 19$	$y \geq 19$	**Maximal**

Existing Work

Non-Vacuous

- ✘ CHC Solvers

Existing Work

Non-Vacuous

- ✘ CHC Solvers

Maximal

- ✘ SyGuS and SMT Solvers

Existing Work

Non-Vacuous

- ✘ CHC Solvers

Maximal

- ✘ SyGuS and SMT Solvers

Complete CHC Solving

- ✘ Maximal Specification Synthesis [POPL'16]

Objective

A technique to find non-vacuous maximal solution to a system of CHCs

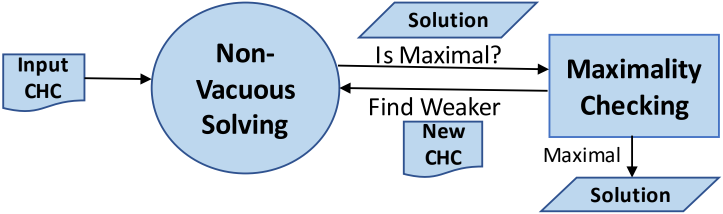
Key Contributions

- **Non-Vacuous CHC Solver**: propagation based algorithm
- **Maximality Checker**: iterative generalization procedure

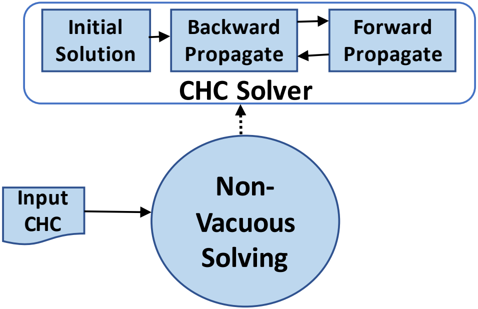
Interlude

- ✓ Specification Synthesis
- ✓ CHCs and constrain on its solutions
- ✓ The need for an algorithm
- ? Algorithm Illustration
- ? Experiment Summary

Algorithm Overview



Non-Vacuous CHC Solver - Overview



- Backward (Forward) Propagation: new candidates for LHS based on RHS (vice versa)

Non-Vacuous CHC Solver - Illustration

$\mathsf{inv}(x) \mapsto \top$
$\mathsf{f}(y) \mapsto \top$

Current Interpretation

$$x = 19 \implies \mathsf{inv}(x)$$

$$\mathsf{inv}(x) \wedge x' = x - 1 \implies \mathsf{inv}(x')$$

$$\mathsf{inv}(x) \wedge \mathsf{f}(y) \implies x \leq y$$

Current CHC

- Uses *Backward Propagation* based on multi-abduction [POPL'16]
- Gets $\mathsf{inv}(x) \mapsto x \leq 0$ and $\mathsf{f}(y) \mapsto y \geq 0$

Non-Vacuous CHC Solver - Illustration

$$\text{inv}(x) \mapsto x \leq 0$$
$$\text{f}(y) \mapsto y \geq 0$$

Current Interpretation

$$x = 19 \implies \text{inv}(x)$$

$$\text{inv}(x) \land x' = x - 1 \implies \text{inv}(x')$$

$$\text{inv}(x) \land \text{f}(y) \implies x \leq y$$

Current CHC

- Failure! Changes the propagation direction
- Uses *Forward Propagation*
- Gets $\text{inv}(x) \mapsto x \leq 19 \land x \geq 19$

Non-Vacuous CHC Solver - Illustration

$$\mathrm{inv}(x) \mapsto x \leq 19 \wedge x \geq 19$$
$$\mathrm{f}(y) \mapsto y \geq 0$$

Current Interpretation

$$x = 19 \implies \mathrm{inv}(x)$$
$$\mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$$
$$\mathrm{inv}(x) \wedge \mathrm{f}(y) \implies x \leq y$$

Current CHC

- Using *Houdini* [FME'01] learns which conjunct is inductive
- Gets $\mathrm{inv}(x) \mapsto x \leq 19$

Non-Vacuous CHC Solver - Illustration

$\mathrm{inv}(x) \mapsto x \leq 19$
$\mathrm{f}(y) \mapsto y \geq 0$

Current Interpretation

$x = 19 \implies \mathrm{inv}(x)$

$\mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$

$\mathrm{inv}(x) \wedge \mathrm{f}(y) \implies x \leq y$

Current CHC

- Backward propagation may give back $\mathrm{inv}(x) \mapsto x \leq 0$ and $\mathrm{f}(y) \mapsto y \geq 0$
- So, uses *Fairness Heuristic*

Non-Vacuous CHC Solver - Illustration

$$\text{inv}(x) \mapsto x \leq 19$$
$$\text{f}(y) \mapsto y \geq 0$$

Current Interpretation

$$x = 19 \implies \text{inv}(x)$$

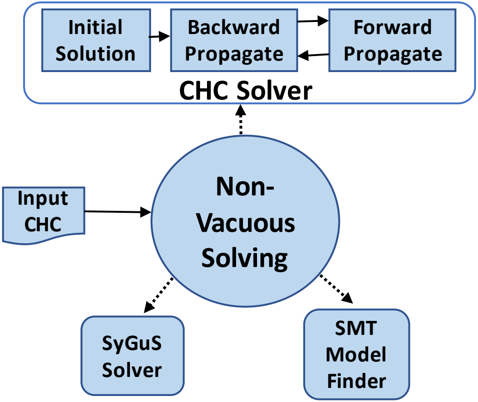
$$\text{inv}(x) \wedge x' = x - 1 \implies \text{inv}(x')$$

$$x \leq 19 \wedge \text{f}(y) \implies x \leq y$$

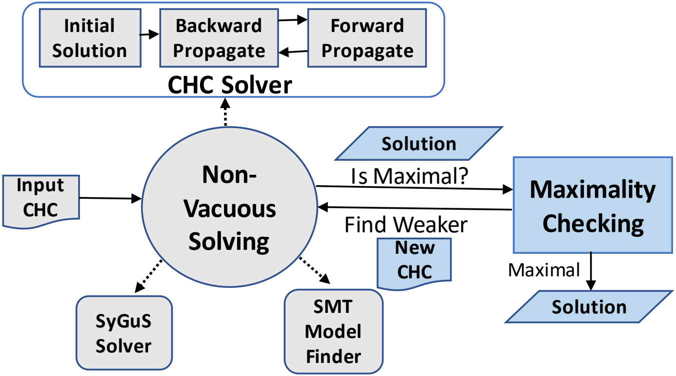
Current CHC

- Chooses relations to fix (here, inv)
- Now, backward propagates to the rest
- Gets non-vacuous solution: $\text{f}(y) \mapsto y \geq 19$

Non-Vacuous Solving - Extension



Maximality Checking



Maximality Checking - Definition

Recall:

- Given:

  S (a system of CHCs)
  R (a set of relations)

- M is *maximal* if no solution M' satisfies

$$\forall r \in R \,.\, M(r) \implies M'(r)$$

and

$$\exists r \in R \,.\, M'(r) \not\Rightarrow M(r)$$

Maximality Checking - Illustration

$$\mathrm{inv}(x) \mapsto x \leq 19$$
$$\mathrm{f}(y) \mapsto y = 19$$

Non-Vacuous Solution

$$x = 19 \implies \mathrm{inv}(x)$$
$$\mathrm{inv}(x) \land x' = x - 1 \implies \mathrm{inv}(x')$$
$$\mathrm{inv}(x) \land \mathrm{f}(y) \implies x \leq y$$

Input CHC

- Intuition: Try to weaken interpretations by at least one more point by adding two conjuncts

Maximality Checking - Illustration

$$\text{inv}(x) \mapsto x \leq 19$$
$$\text{f}(y) \mapsto y = 19$$

Non-Vacuous Solution

$$x = 19 \implies \text{inv}(x)$$
$$\text{inv}(x) \wedge x' = x - 1 \implies \text{inv}(x')$$
$$\text{inv}(x) \wedge \text{f}(y) \implies x \leq y$$

Input CHC

1. In input CHC, substitute

$$\text{inv}(x) \mapsto x \leq 19 \vee x = p_x$$
$$\text{f}(y) \mapsto y = 19 \vee y = p_y$$

Maximality Checking - Illustration

$\mathsf{inv}(x) \mapsto x \leq 19$
$\mathsf{f}(y) \mapsto y = 19$

Non-Vacuous Solution

$x = 19 \implies \mathsf{inv}(x)$

$\mathsf{inv}(x) \wedge x' = x - 1 \implies \mathsf{inv}(x')$

$\mathsf{inv}(x) \wedge \mathsf{f}(y) \implies x \leq y$

Input CHC

2. Constrain values of placeholder variables p_x, p_y

$$\neg(p_x \leq 19) \vee \neg(p_y = 19)$$

Maximality Checking - Illustration

$\mathrm{inv}(x) \mapsto x \leq 19$
$\mathrm{f}(y) \mapsto y = 19$

Non-Vacuous Solution

$x = 19 \implies \mathrm{inv}(x)$

$\mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$

$\mathrm{inv}(x) \wedge \mathrm{f}(y) \implies x \leq y$

Input CHC

- $CTM \models 1 \wedge 2$
- Based on values of p_x and p_y from *counterexample-to-maximality* (CTM), decide relations to weaken

Maximality Checking - Illustration

$$x = 19 \implies \mathsf{inv}(x)$$

$$\mathsf{inv}(x) \wedge x' = x - 1 \implies \mathsf{inv}(x')$$

$$\mathsf{inv}(x) \wedge \mathsf{f}(y) \wedge \implies x \leq y$$

$$y = 19 \implies \mathsf{f}(y)$$

$$\neg(y = 19) \wedge \mathsf{p_f}(y) \implies \mathsf{f}(y)$$

New CHCs for Weakening

- A non-vacuous solution to $\mathsf{p_f}$ ensures that current solution for $M(\mathsf{f})$ is weakened

Maximality Checking - Illustration

$$x = 19 \implies \mathrm{inv}(x)$$

$$\mathrm{inv}(x) \wedge x' = x - 1 \implies \mathrm{inv}(x')$$

$$\mathrm{inv}(x) \wedge \mathrm{f}(y) \wedge \implies x \leq y$$

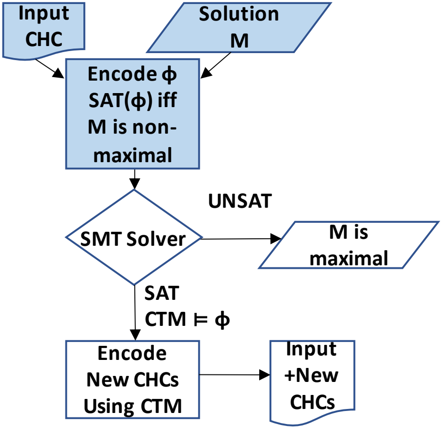
$$y = 19 \implies \mathrm{f}(y)$$

$$\neg(y = 19) \wedge \mathrm{p_f}(y) \implies \mathrm{f}(y)$$

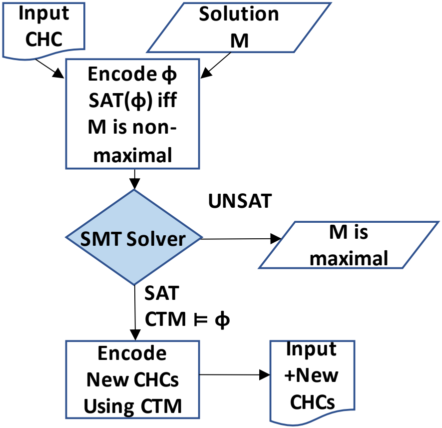
New CHCs for Weakening

- $\mathrm{p_f}(y) \mapsto y = 20$ and $\mathrm{f}(y) \mapsto y \geq 19$

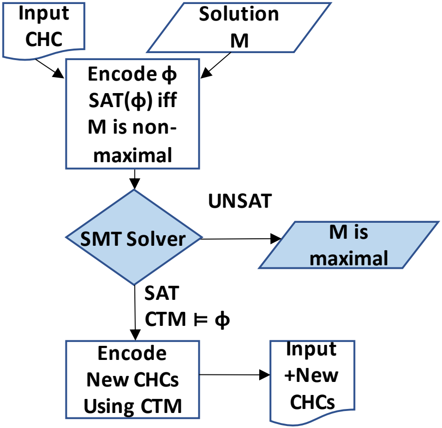
Maximality Checking - Overview



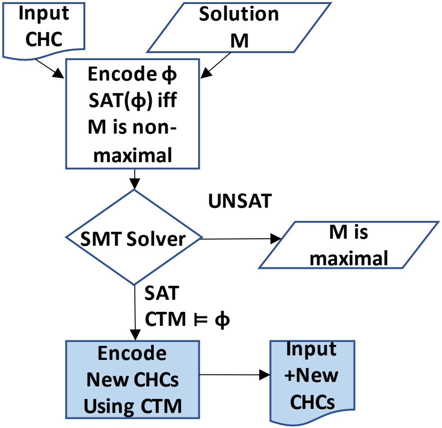
Maximality Checking - Overview



Maximality Checking - Overview



Maximality Checking - Overview



Experiment Goals

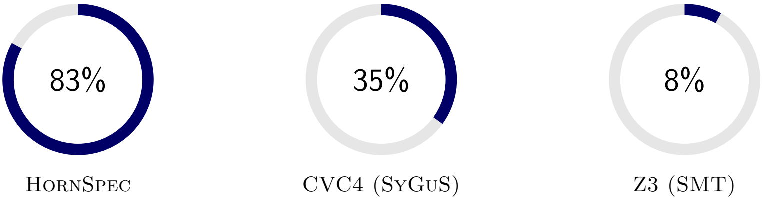
1. Can the technique generate maximal solutions reasonably quickly?
2. Does the non-vacuous solving help in the performance?

Experiment Setup

- Tool: HornSpec built on top of FreqHorn [FMCAD'18] framework
- Supports non-vacuous solving using CVC4 (SyGuS) and Z3 (SMT) solvers
- Benchmarks: 65 CHC systems in LIA majorly from CHC-Comp

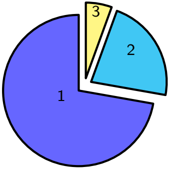
Experiment Summary

Maximal Solutions



Experiment Summary

#Iterations to extend non-vacuous to maximal



- Non-Vacuous solutions generated by HORNSPEC were *almost* maximal

Experiment Summary

- Time taken less than a minute

  HornSpec 54/54

  CVC4 20/22

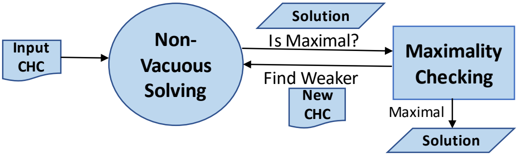
  Z3 4/5

- HornSpec outperformed in majority of benchmarks solved
- On no benchmarks CVC4 or Z3 was able to find a maximal specification, but HornSpec could not

Conclusion

A technique to find non-vacuous and maximal solution to a system of CHCs

- Non-Vacuous CHC Solver
- Maximality Checker



Paper (free access) at `https://doi.org/10.1145/3453483.3454104`

References

POPL'16 Aws Albarghouthi, Isil Dillig, and Arie Gurfinkel, Maximal Specification Synthesis, POPL'16

FME'01 Cormac Flanagan and K. Rustan M. Leino, Houdini: an Annotation Assistant for ESC/Java, FME'01

FMCAD'18 Grigory Fedyukovich, Sumanth Prabhu, Kumar Madhukar, and Aarti Gupta, Solving Constrained Horn Clauses Using Syntax and Data, FMCAD'18

Backup

Quality Solutions to CHCs

- Given

  S (a system of CHCs)
  R (a set of relations)

- A solution M to S is *vacuous* if

  $\exists \mathrm{r} \in R \, . \, M(\mathrm{r}) \implies \bot$

  or

  $\exists C \in S \, . \, \neg query(C) \wedge lhs(C)[M] \implies \bot$

Quality Solutions to CHCs

- Given

  S (a system of CHCs)
  R (a set of relations)

- M is *maximal* if no solution M' satisfies

$$\forall \mathrm{r} \in R \,.\, M(\mathrm{r}) \implies M'(\mathrm{r})$$

and

$$\exists \mathrm{r} \in R \,.\, M'(\mathrm{r}) \not\Longrightarrow M(\mathrm{r})$$