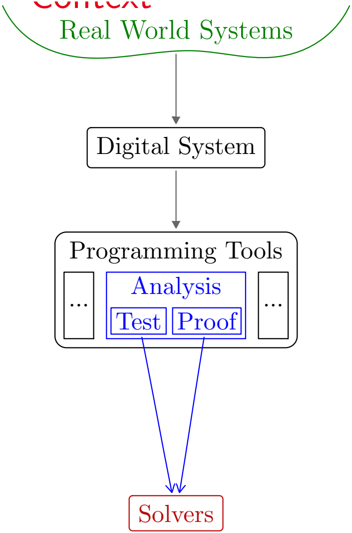
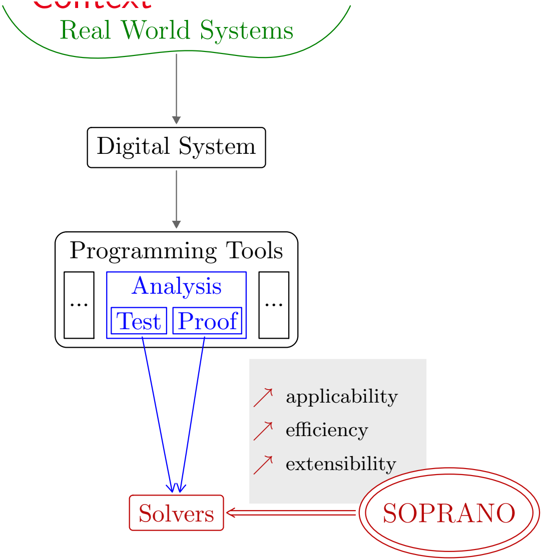
Context

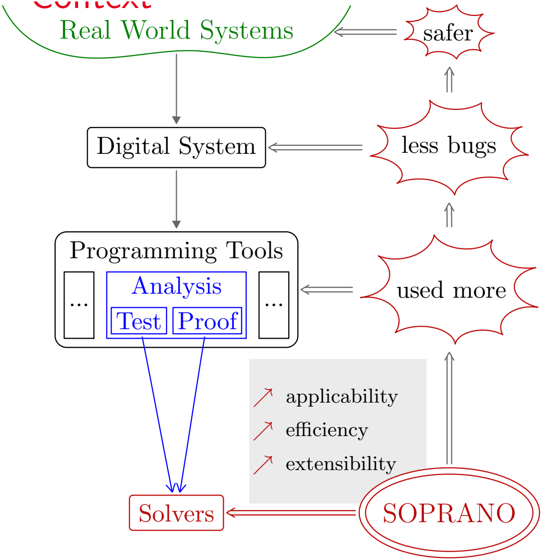


Real World Systems
Digital System
Programming Tools
...
Analysis
Test
Proof
...
Solvers

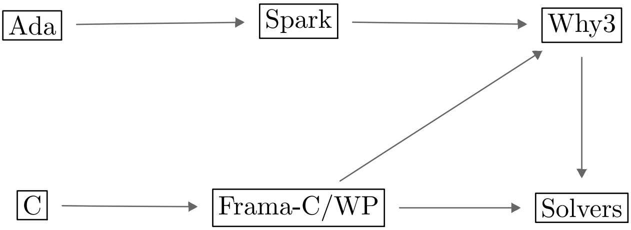
Context


Real World Systems
Digital System
Programming Tools
...
Analysis
Test
Proof
...
applicability
efficiency
extensibility
Solvers
SOPRANO

Context


Real World Systems
safer
Digital System
less bugs
Programming Tools
...
Analysis
Test
Proof
...
used more
applicability
efficiency
extensibility
Solvers
SOPRANO

Toolchains



Challenges

Currently uses SMT solvers:

- ✔ Good handling of arithmetic (integers, bitvectors, reals)
- ✔ Good handling of axioms
- ✔ Reasonably fast
- ✔ Agreed semantics between all provers (SMTLIB)

Challenges

Currently uses SMT solvers:

- ✔ Good handling of arithmetic (integers, bitvectors, reals)
- ✔ Good handling of axioms
- ✔ Reasonably fast
- ✔ Agreed semantics between all provers (SMTLIB)

- ✘ Many properties involving x/y, $x \times y$, x^y, $x \bmod y$, $x \operatorname{rem} y$
- ✘ Most properties involving floating-point values
- ✘ Properties involving conversions between types (integers $\leftrightarrow$ bitvectors, integers $\leftrightarrow$ reals, integers $\leftrightarrow$ floats)

Floating Points

- ✔ Clear Semantic: $x \oplus y = \circ(x + y)$
- ✘ Few algebraic properties: not associative, $x \oplus y = x \not\Rightarrow y = 0$
- ✘ Counter-intuitive: $\overbrace{0.1 \oplus \cdots \oplus 0.1}^{10} \neq 0.1 \otimes 10. = 1.$
- ✘ State of the art: current bit-blasting doesn't scale
- ✘ Pervasives in programs

Interesting and Simple Real Examples

```
/*@ requires 0 ≤ x ≤ 1000;
    requires 0 ≤ y ≤ 1000;
    ensures  0 ≤ \result ≤ 1;   @*/
double x_normalisation(double x,double y){

  return x/sqrt(x*x + y*y);

}
```

Domain Specific Approach of CP

$$X_i \in [1;10] \implies X_0 \oplus X_1 \oplus X_2 \oplus X_3 \oplus X_4 \oplus X_5 \oplus X_6 \oplus X_7 \in [8;80]$$

Z3 (SMT): 3s
COLIBRI (CP): 0.1s

Domain Specific Approach of CP

$$X_i \in [1;10] \implies X_0 \oplus X_1 \oplus X_2 \oplus X_3 \oplus X_4 \oplus X_5 \oplus X_6 \oplus X_7 \in [8;80]$$

Z3 (SMT): 3s
COLIBRI (CP): 0.1s

$$X_i \in [1;10] \implies X_0 \otimes X_1 \otimes X_2 \otimes X_3 \otimes X_4 \otimes X_5 \otimes X_6 \otimes X_7 \in [1;10^8]$$

Z3 (SMT): 31min
COLIBRI (CP): 0.1s

COLIBRI: Floating Point (Bruno Marre)

- Precise domain propagation:
  $x \oplus y = 0.05 \implies x, y \in [-0.1259..; 0.175....]$

COLIBRI: Floating Point (Bruno Marre)

- Precise domain propagation:
  $x \oplus y = 0.05 \implies x, y \in [-0.1259..; 0.175....]$
- Distance graph on floating-point numbers

COLIBRI: Floating Point (Bruno Marre)

- Precise domain propagation:
  $x \oplus y = 0.05 \implies x, y \in [-0.1259..; 0.175....]$
- Distance graph on floating-point numbers
- Monotonic functions:
  $o(f(x)) < o(y) \implies o(x) \leq o(f^{-1}(o(y)))$

COLIBRI: Floating Point (Bruno Marre)

- Precise domain propagation:
  $x \oplus y = 0.05 \implies x, y \in [-0.1259..; 0.175....]$
- Distance graph on floating-point numbers
- Monotonic functions:
  $\mathrm{o}(f(x)) < \mathrm{o}(y) \implies \mathrm{o}(x) \leq \mathrm{o}(f^{-1}(\mathrm{o}(y)))$
- Instantiated for many functions

COLIBRI: Floating Point (Bruno Marre)

- Precise domain propagation:
  $x \oplus y = 0.05 \implies x, y \in [-0.1259..; 0.175....]$
- Distance graph on floating-point numbers
- Monotonic functions:
  $\mathrm{o}(f(x)) < \mathrm{o}(y) \implies \mathrm{o}(x) \leq \mathrm{o}(f^{-1}(\mathrm{o}(y)))$
- Instantiated for many functions
- Linearization of constraints for simplex

COLIBRI: Example of Reasoning

$$0 \leq x, y \leq 1000 \implies \sqrt{x^2 \oplus y^2} \geq x \ ?$$

COLIBRI: Example of Reasoning

$$0 \leq x, y \leq 1000 \implies \sqrt{x^2 \oplus y^2} \geq x \ ?$$

$$\mathrm{o}\left(\sqrt{\mathrm{o}(x^2) + \mathrm{o}(y^2)}\right) < x$$

$$\mathrm{o}(x^2) + \mathrm{o}(y^2) \leq \mathrm{o}(x^2)$$

$$\mathrm{o}(x^2) + \mathrm{o}(y^2) = \mathrm{o}(x^2)$$

$$\mathrm{o}\left(\sqrt{\mathrm{o}(x^2)}\right) < x$$

$x < x$ if $\mathrm{o}(x^2)$ is normalized

$\mathrm{o}(x^2)$ is denormalized

x the minimum of the remaining values is a solution

COLIBRI: Example of Reasoning

$$0 \leq x, y \leq 1000 \implies \sqrt{x^2 \oplus y^2} \geq x \; ?$$

$$\mathrm{o}\left(\sqrt{\mathrm{o}(x^2) + \mathrm{o}(y^2)}\right) < x$$

$$\mathrm{o}(x^2) + \mathrm{o}(y^2) \leq \mathrm{o}(x^2)$$

$$\mathrm{o}(x^2) + \mathrm{o}(y^2) = \mathrm{o}(x^2)$$

$$\mathrm{o}\left(\sqrt{\mathrm{o}(x^2)}\right) < x$$

$x < x$ if $\mathrm{o}(x^2)$ is normalized

$\mathrm{o}(x^2)$ is denormalized

x the minimum of the remaining values is a solution

There is a counter-example!

Interesting and Simple Real Examples: Corrected

```
/*@ requires 0.0001 ≤ x ≤ 1000;
    requires 0.0001 ≤ y ≤ 1000;
    ensures  0 ≤ \result ≤ 1;    @*/
double x_normalisation(double x,double y){

  return x/sqrt(x*x + y*y);

}
```

Other Examples: From SPARK User Rule

```
procedure User_Rule_7 (X, Y, Z, A : Float;
                       Res        : out Boolean)
 is
 begin
    pragma Assume (Z ≥ 0.0);
    pragma Assume (X ≥ Y);
    pragma Assume (Y ≥ Z);
    pragma Assume (X > Z);
    pragma Assume (A ≥ 1.0);
    Res := (X − Y) / (X − Z) ≤ A;
    pragma Assert (Res);      −− valid
 end User_Rule_7;
```

Other Examples: From SPARK User Rule

$$A \leq \frac{X \ominus Y}{X \ominus Z} \leq B \quad \text{with} \dots$$

$$\sqrt{X^2 \ominus Y^2} \leq X \quad \text{with} \dots$$

$$\frac{X}{\sqrt{X^2 \oplus Y^2}} \leq 1 \quad \text{with} \dots$$

Floating-Point Arithmetic: Alt-Ergo
(OCamlPro, UPSud)

Floating-point rounding operator on rational constants; Interval domains

```
axiom rounding_operator_1 :
   forall  x :  real .
   forall  i , j  :  real .
   forall  md : fpa_rounding_mode.
   forall  p,m :  int
    [round(m,p,md,x), x in  [i , j ]].
      i  ≤ x ≤ j →
      round(m,p,md,i) ≤ round(m,p,md,x) ≤ round(m,p,md,j)
```

(Kailiang Ji, post-doc SOPRANO)

Floating-Point Arithmetic: Recap

Progression of COLIBRI and Alt-Ergo on AdaCore benchmarks:

	Before SOPRANO	Current Results
COLIBRI	18 / 28	25 / 28
Alt-Ergo	2 / 28	19 / 28

Bitvector and Integer Arithmetic (Chihani)

COLIBRI:

- High-level view of bitvectors
- New propagations for integers $\leftrightarrow$ bitvectors

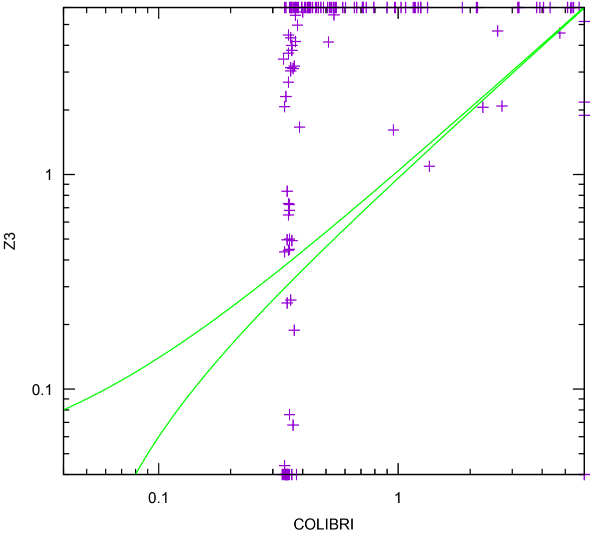
Evaluations & Diffusion

- AdaCore examples

Evaluations & Diffusion

- AdaCore examples
- Participation at SMT-COMP 2017

COLIBRI on SMT-COMP FP category



Evaluations & Diffusion

- AdaCore examples
- Participation at SMT-COMP 2017

Evaluations & Diffusion

- AdaCore examples
- Participation at SMT-COMP 2017

COLIBRI: Freeware For Research

F. Bobot, Z. Chihani, M. Iguernlala, and B. Marre.
Fpa solver.
Technical report, ANR SOPRANO, ANR-14-CE28-0020, 2016.
http://soprano-project.fr/downloads/D3_1.pdf.

Z. Chihani, B. Marre, F. Bobot, and S. Bardin.
Sharpening constraint programming approaches for bit-vector theory.
In *CPAIOR*, 2017.

S. Conchon, M. Iguernlala, K. Ji, G. Melquiond, and C. Fumex.
A three-tier strategy for reasoning about floating-point numbers in smt.
In *Computer Aided Verification*, 2017.

Floating-Point Arithmetic: Monotonic functions (CEA, UPSud)

Theorem

Let $D, E \subset \mathcal{R}$, $f : D \mapsto E$ and $f^{-1} : E \mapsto D$ such that

- $\forall x : D,\ f^{-1}(f(x)) = x$
- *f increasing*

We have

- $\forall x \in D, \mathrm{o}(y) \in E, \mathrm{o}(f(x)) < \mathrm{o}(y) \implies \mathrm{o}(x) \leq \mathrm{o}(f^{-1}(\mathrm{o}(y)))$
- $\forall x \in D, y \in E, \mathrm{o}(f(x)) < \mathrm{o}(f(y)) \implies x < y$

Instantiated for many functions in COLIBRI's DBM

Interesting and Simple Real Examples

```
/*@ ensures \result ≤ (double) 1; @*/
double test2(){
  double x = read_sensor();
  /*@ assert (double) 0 ≤ x ≤ (double) 1000; @*/
  double y = read_sensor();
  double z = read_sensor();

  x = x * x + z * z + y * y + 1;

  if (z ≤ y){
    return (x−y)/(x−z);
  } else {
    return (x−z)/(x−y);
  }
}
```

The Problems