

SPASS+T

Uwe Waldmann

MPI für Informatik, Saarbrücken, Germany

SPASS+T

Pragmatic integration of arithmetic knowledge into SPASS:

SPASS+T

Pragmatic integration of arithmetic knowledge into SPASS:

Superposition-based ATP

SPASS+T

Pragmatic integration of arithmetic knowledge into SPASS:

Superposition-based ATP

+ standard axioms

$$\forall x, y. (x + (-y)) + y = x$$

SPASS+T

Pragmatic integration of arithmetic knowledge into SPASS:

Superposition-based ATP

+ standard axioms

$$\forall x, y. (x + (-y)) + y = x$$

+ built-in arithmetic simplification and inference rules

$$\frac{C \vee t + c_1 < c_2}{C \vee t < c_0} \quad \text{where } c_0 = c_2 - c_1$$

SPASS+T

Pragmatic integration of arithmetic knowledge into SPASS:

Superposition-based ATP

+ standard axioms

$$\forall x, y. (x + (-y)) + y = x$$

+ built-in arithmetic simplification and inference rules

$$\frac{C \vee t + c_1 < c_2}{C \vee t < c_0} \quad \text{where } c_0 = c_2 - c_1$$

+ external SMT procedure (currently: Yices or CVC3)

pass input and derived clauses to SMT,
treat SMT refutations like SPASS inferences

SPASS+T

Thanks to

SPASS+T

Thanks to

- Christoph Weidenbach
for **SPASS**

SPASS+T

Thanks to

- Christoph Weidenbach
for **SPASS**
- Bruno Dutertre, Leonardo de Moura
for **T** (in CASC-23: Yices 1.0.29)

SPASS+T

Thanks to

- Christoph Weidenbach
for **SPASS**
- Bruno Dutertre, Leonardo de Moura
for **T** (in CASC-23: Yices 1.0.29)
- Stephan Zimmer
for a substantial part of **+**

SPASS+T

Thanks to

- Christoph Weidenbach
for **SPASS**
- Bruno Dutertre, Leonardo de Moura
for **T** (in CASC-23: Yices 1.0.29)
- Stephan Zimmer
for a substantial part of **+**

(OK, that doesn't leave much for me ...)

SPASS+T

Thanks to

SPASS+T

Thanks to

- Silvio Ranise

for sending a collection of problems from *Manna and Pnueli: Tools and Rules for the Practicing Verifier* to Geoff Sutcliffe

SPASS+T

Thanks to

- Silvio Ranise

for sending a collection of problems from *Manna and Pnueli: Tools and Rules for the Practicing Verifier* to Geoff Sutcliffe

~> Fixpoint problems in the TPTP SWW domain

- seem to be very hard for Z3
(due to peculiarities of the encoding?)

SPASS+T

Thanks to

- Silvio Ranise

for sending a collection of problems from *Manna and Pnueli: Tools and Rules for the Practicing Verifier* to Geoff Sutcliffe

↪ Fixpoint problems in the TPTP SWW domain

- seem to be very hard for Z3
(due to peculiarities of the encoding?)
- SPASS+T finds proofs using superposition and chaining
(inferences: ~ 10000 , proof length: ~ 1000).

SPASS+T

Second type of problems where SPASS+T has an edge:

$\forall a.$

$(\forall i. 20 \leq i \wedge i \leq 30 \rightarrow read(a, i) = i + 25)$

\rightarrow

$\exists j. read(a, j) = 50.$

SPASS+T

Second type of problems where SPASS+T has an edge:

$\forall a.$

$(\forall i. 20 \leq i \wedge i \leq 30 \rightarrow read(a, i) = i + 25)$

\rightarrow

$\exists j. read(a, j) = 50.$

But these are intuitively easy:

One superposition step + (trivial) variable elimination.

Why does Z3 fail here?

SPASS+T

Conclusions:

- There are many techniques to solve arithmetic problems (DPLL(T), trigger-based instantiation, variable elimination, superposition, chaining, ...).

SPASS+T

Conclusions:

- There are many techniques to solve arithmetic problems (DPLL(T), trigger-based instantiation, variable elimination, superposition, chaining, ...).
- There are several ways to combine these techniques.

SPASS+T

Conclusions:

- There are many techniques to solve arithmetic problems (DPLL(T), trigger-based instantiation, variable elimination, superposition, chaining, ...).
- There are several ways to combine these techniques.
- There is still a lot of room for improvements.

SPASS+T

Conclusions:

- There are many techniques to solve arithmetic problems (DPLL(T), trigger-based instantiation, variable elimination, superposition, chaining, ...).
- There are several ways to combine these techniques.
- There is still a lot of room for improvements.
- By the way:
We need more interesting TFA problems in the TPTP.

SPASS+T

CASC-23 TFA division: 96/100 problems solved.

SMT	Std	Built-in	#Problems
Refut	Axioms	Arithm	
+	+	+	2
+	+	-	0
+	-	+	1
+	-	-	15
-	+	+	26
-	+	-	4
-	-	+	48

Caveat: non-deterministic due to unsynchronized parallelism.