FONDAZIONE BRUNO KESSLER

Deduction at Scale Seminar 2011

Efficient Interpolant Generation in Satisfiability Modulo Linear Integer Arithmetic

Alberto Griggio

FBK-IRST, Trento

joint work with Thi Thieu Hoa Le and Roberto Sebastiani, DISI - Univ. Trento

Introduction

- (Craig) Interpolation for ground first-order theories successfully applied in formal verification
  - Efficient SMT-based algorithms for several theories and combinations (e.g. EUF, LA(Q), DL, UTVPI)
- Interpolation for full LA(Z) is harder
  - Some promising recent work [Brillout et al IJCAR'10, Kroening et al. LPAR'10], but still some drawbacks
- This work: propose a novel, general technique for interpolation in LA(Z)
  - to overcome some drawbacks of current approaches

Outline

- Background
- Current techniques for interpolation in LA(Z)
- A novel interpolation technique for LA(Z)
- Experimental evaluation

Background - Interpolants

- (Craig) Interpolant for an ordered pair (A, B) of formulas s.t.

  $A \wedge B \models_{\mathcal{T}} \bot$ is a formula I s.t.

  a) $A \models_{\mathcal{T}} I$

  b) $B \wedge I \models_{\mathcal{T}} \bot$

  c) all the uninterpreted (in $\mathcal{T}$) symbols of I occur in both A and B

Background - Interpolants

- Interpolants can be generated from proofs of unsatisfiability [McMillan]

Background - Interpolants

- Interpolants can be generated from proofs of unsatisfiability [McMillan]
- Proof of unsatisfiability in SMT:

Boolean part (ground resolution)	+	$\mathcal{T}$-specific part (for conjunctions of constraints)

Background - Interpolants

- Interpolants can be generated from proofs of unsatisfiability [McMillan]
- Proof of unsatisfiability in SMT:

Boolean part
(ground resolution)

+

$\mathcal{T}$-specific part
(for conjunctions of constraints)

Standard Boolean interpolation

$\mathcal{T}$-specific
interpolation
for conjunctions only

Background - Interpolants

- Interpolants can be generated from proofs of unsatisfiability [McMillan]
- Proof of unsatisfiability in SMT:

Boolean part
(ground resolution)

+

$\mathcal{T}$-specific part
(for conjunctions of constraints)

Standard Boolean interpolation

$\mathcal{T}$-specific interpolation for conjunctions only

Problem reduced to finding an interpolant for sets of $\mathcal{T}$-literals

Outline

- Background
- Current techniques for interpolation in LA(Z)
- A novel interpolation technique for LA(Z)
- Experimental evaluation

Interpolation and LA(Z)

FONDAZIONE BRUNO KESSLER

- Linear Integer Arithmetic: constraints of the form

$$\sum_i c_i x_i + c \bowtie 0, \qquad \bowtie \in \{\leq, =\}$$

- In general, no quantifier-free interpolation for LA(Z)! [McMillan05]

Example: $A := (y - 2x = 0) \quad B := (y - 2z - 1 = 0)$

The only interpolant is: $\exists w.(y = 2w)$

- Solution: extend the signature to include modular equations (divisibility predicates)

$$(t + c =_d 0) \equiv \exists w.(t + c = d \cdot w), \quad d \in \mathbb{Z}^{>0}$$

The interpolant now becomes: $(y =_2 0)$

SMT(LA(Z)) with modular equations

- Modular equations can be eliminated via preprocessing:
  - Replace every atom $a := (t + c =_d 0)$ with a fresh Boolean variable p_a
  - Add the 4 clauses

$$p_a \rightarrow (t + c - dw_1 = 0)$$
$$\neg p_a \rightarrow (t + c - dw_1 - w_2 = 0)$$
$$(-w_2 + 1 \leq 0)$$
$$(w_2 - d + 1 \leq 0)$$

    where w_1, w_2 are fresh integer variables

Interpolation via quantifier elimination

- Using modular equation, interpolants can be constructed via quantifier elimination:

$$I(A, B) := \mathrm{ExistElim}(x_i \not\in B)(A)$$

- However, this is very expensive, both in theory and in practice

Interpolants from LA(Z)-proofs

- Cutting-plane proof system: complete proof system for LA(Z)

$$\text{Hyp}\ \frac{-}{t \leq 0} \qquad \text{Comb}\ \frac{t_1 \leq 0 \qquad t_2 \leq 0}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0}, c_1, c_2 > 0$$

$$\text{Div}\ \frac{\sum_i c_i x_i + c \leq 0}{\sum_i \frac{c_i}{d} x_i + \lceil \frac{c}{d} \rceil \leq 0}, d > 0 \text{ divides the } c_i\text{'s}$$

Interpolants from LA(Z)-proofs

- Cutting-plane proof system: complete proof system for LA(Z)

$$\text{Hyp}\ \frac{-}{t \leq 0} \qquad \text{Comb}\ \frac{t_1 \leq 0 \qquad t_2 \leq 0}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0}, c_1, c_2 > 0$$

LA(Q) rules

$$\text{Div}\ \frac{\sum_i c_i x_i + c \leq 0}{\sum_i \frac{c_i}{d} x_i + \lceil \frac{c}{d} \rceil \leq 0}, d > 0 \text{ divides the } c_i\text{'s}$$

Interpolants from LA(Z)-proofs

- Cutting-plane proof system: complete proof system for LA(Z)

$$\text{Hyp}\ \frac{-}{t \leq 0} \qquad \text{Comb}\ \frac{t_1 \leq 0 \qquad t_2 \leq 0}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0}, c_1, c_2 > 0$$

$$\text{Strenghten}\ \frac{\sum_i c_i x_i + c \leq 0}{\sum_i c_i x_i + d \cdot \lceil \frac{c}{d} \rceil \leq 0}, d > 0 \text{ divides the } c_i\text{'s}$$

Interpolants from LA(Z)-proofs

- Cutting-plane proof system: complete proof system for LA(Z)

$$\text{Hyp}\ \frac{-}{t \leq 0} \qquad\qquad \text{Comb}\ \frac{t_1 \leq 0 \qquad t_2 \leq 0}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0}, c_1, c_2 > 0$$

$$\text{Strenghten}\ \frac{\sum_i c_i x_i + c \leq 0}{\sum_i c_i x_i + d \cdot \lceil \frac{c}{d} \rceil \leq 0}, d > 0 \text{ divides the } c_i\text{'s}$$

- Interpolation by annotating proof rules [McMillan05, Brillout et al. IJCAR'10]
  - Annotation (in this talk): a set of pairs $\{\langle t_i \leq 0, \bigwedge_j (t_{ij} = 0)\rangle\}_i$
  - When $\perp$ is derived, then

$$I := \bigvee_i (t_i \leq 0 \wedge \bigwedge_j \text{ExistElim}(x_i \notin B).(t_{ij} = 0))$$

  is the computed interpolant

Interpolants from cutting-plane proofs

- Annotations for Hyp and Comb from [McMillan05] (same as LA(Q))

$$\text{Hyp}\ \frac{-}{t \leq 0\ [\{\langle t' \leq 0, \top\rangle\}]}\quad t' = \begin{cases} t & \text{if } t \leq 0 \in A \\ 0 & \text{if } t \leq 0 \in B \end{cases}$$

$$\text{Comb}\ \frac{t_1 \leq 0\ [I_1] \qquad t_2 \leq 0\ [I_2]}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0\ [I]}$$

$$I := \{\langle c_1 t_i' + c_2 t_j' \leq 0, E_i \wedge E_j\rangle \mid \langle t_i', E_i\rangle \in I_1, \langle t_j', E_j\rangle \in I_2\}$$

- k-Strengthen rule of [Brillout et al. IJCAR'10] (special case)

$$\text{Str.}\ \frac{\sum_i c_i x_i + c \leq 0\ [\{\langle t \leq 0, \top\rangle\}]}{\sum_i c_i x_i + d \cdot \lceil \frac{c}{d} \rceil \leq 0\ [I]}, d > 0 \text{ divides the } c_i\text{'s}$$

$$I := \{\langle (t + n \leq 0),\ (t + n = 0)\rangle \mid 0 \leq n < d \cdot \lceil \tfrac{c}{d} \rceil - c\} \cup \{\langle (t + d \cdot \lceil \tfrac{c}{d} \rceil - c \leq 0), \top\rangle\}$$

Interpolants from cutting-plane proofs

- Annotations for Hyp and Comb from [McMillan05] (same as LA(Q))

$$\text{Hyp}\ \frac{-}{t \leq 0\ [\{\boxed{\langle t \leq 0,} \top\rangle\}]} \quad t' = \begin{cases} t & \text{if } t \leq 0 \in A \\ 0 & \text{if } t \leq 0 \in B \end{cases}$$

$$\text{Comb}\ \frac{t_1 \leq 0\ [I_1] \qquad t_2 \leq 0\ [I_2]}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0\ [I]}$$

$$I := \{\langle c_1 t'_i + c_2 t'_j \leq 0, E_i \wedge E_j\rangle \mid \langle t'_i, E_i\rangle \in I_1, \langle t'_j, E_j\rangle \in I_2\}$$

- k-Strengthen rule of [Brillout et al. IJCAR'10] (special case)

$$\text{Str.}\ \frac{\sum_i c_i x_i + c \leq 0\ [\{\langle t \leq 0, \top\rangle\}]}{\sum_i c_i x_i + d \cdot \lceil \frac{c}{d} \rceil \leq 0\ [I]}, d > 0 \text{ divides the } c_i\text{'s}$$

$$I := \{\langle (t + n \leq 0),\ (t + n = 0)\rangle \mid 0 \leq n < d \cdot \lceil \tfrac{c}{d} \rceil - c\} \cup \{\langle (t + d \cdot \lceil \tfrac{c}{d} \rceil - c \leq 0), \top\rangle\}$$

Interpolants from cutting-plane proofs

- Annotations for Hyp and Comb from [McMillan05] (same as LA(Q))

$$\text{Hyp}\ \frac{-}{t \leq 0\ [\{\langle \boxed{0 \leq 0,} \top\rangle\}]} \quad t' = \begin{cases} t & \text{if } t \leq 0 \in A \\ 0 & \text{if } t \leq 0 \in B \end{cases}$$

$$\text{Comb}\ \frac{t_1 \leq 0\ [I_1] \qquad t_2 \leq 0\ [I_2]}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0\ [I]}$$

$$I := \{\langle c_1 t'_i + c_2 t'_j \leq 0, E_i \wedge E_j\rangle \mid \langle t'_i, E_i\rangle \in I_1, \langle t'_j, E_j\rangle \in I_2\}$$

- k-Strengthen rule of [Brillout et al. IJCAR'10] (special case)

$$\text{Str.}\ \frac{\sum_i c_i x_i + c \leq 0\ [\{\langle t \leq 0, \top\rangle\}]}{\sum_i c_i x_i + d \cdot \lceil \frac{c}{d} \rceil \leq 0\ [I]}, d > 0 \text{ divides the } c_i\text{'s}$$

$$I := \{\langle (t + n \leq 0),\ (t + n = 0)\rangle \mid 0 \leq n < d \cdot \lceil \tfrac{c}{d} \rceil - c\} \cup \{\langle (t + d \cdot \lceil \tfrac{c}{d} \rceil - c \leq 0), \top\rangle\}$$

Example [Kroening et al. LPAR'10]

$$A := \begin{cases} -y - 4x - 1 \leq 0 \\ y + 4x \leq 0 \end{cases} \qquad B := \begin{cases} -y - 4z + 1 \leq 0 \\ y + 4z - 2 \leq 0 \end{cases}$$

$$\frac{y + 4x \leq 0 \qquad -y - 4z + 1 \leq 0}{4x - 4z + 1 \leq 0}$$

$$\frac{4x - 4z + 1 \leq 0}{4x - 4z + 1 + 3 \leq 0} \qquad \frac{-y - 4x - 1 \leq 0 \qquad y + 4z - 2 \leq 0}{-4x + 4z - 3 \leq 0}$$

$$\frac{4x - 4z + 1 + 3 \leq 0 \qquad -4x + 4z - 3 \leq 0}{(1 \leq 0) \equiv \bot}$$

Example – with annotations

$$A := \begin{cases} -y - 4x - 1 \leq 0 \\ y + 4x \leq 0 \end{cases} \qquad B := \begin{cases} -y - 4z + 1 \leq 0 \\ y + 4z - 2 \leq 0 \end{cases}$$

$$\cfrac{\cfrac{\begin{array}{cc} y + 4x \leq 0 & -y - 4z + 1 \leq 0 \\ [\{\langle y + 4x \leq 0, \top\rangle\}] & [\{\langle 0 \leq 0, \top\rangle\}] \end{array}}{\begin{array}{c} 4x - 4z + 1 \leq 0 \\ [\{\langle y + 4x \leq 0, \top\rangle\}] \end{array}}}{\begin{array}{c} 4x - 4z + 1 + 3 \leq 0 \\ [\{\langle y + 4x + n \leq 0, y + 4x + n = 0\rangle \mid \\ 0 \leq n < 3\} \cup \{\langle y + 4x + 2 \leq 0, \top\rangle\}] \end{array}} \qquad \cfrac{\begin{array}{cc} -y - 4x - 1 \leq 0 & y + 4z - 2 \leq 0 \\ [\{\langle -y - 4x - 1 \leq 0, \top\rangle\}] & [\{\langle 0 \leq 0, \top\rangle\}] \end{array}}{\begin{array}{c} -4x + 4z - 3 \leq 0 \\ [\{\langle -y - 4x - 1 \leq 0, \top\rangle\}] \end{array}}$$

$$\overline{\begin{array}{c} (1 \leq 0) \equiv \bot \\ [\{\langle n - 1 \leq 0, y + 4x + n = 0\rangle \mid 0 \leq n < 3\} \cup \{\langle 2 - 1 \leq 0, \top\rangle\}] \end{array}}$$

Example – with annotations

$$A := \begin{cases} -y - 4x - 1 \leq 0 \\ y + 4x \leq 0 \end{cases} \qquad B := \begin{cases} -y - 4z + 1 \leq 0 \\ y + 4z - 2 \leq 0 \end{cases}$$

Interpolant:
$(y =_4 0) \vee (y + 1 =_4 0)$

$$\dfrac{\dfrac{\dfrac{\begin{matrix} y + 4x \leq 0 & -y - 4z + 1 \leq 0 \\ [\{\langle y + 4x \leq 0, \top\rangle\}] & [\{\langle 0 \leq 0, \top\rangle\}] \end{matrix}}{\begin{matrix} 4x - 4z + 1 \leq 0 \\ [\{\langle y + 4x \leq 0, \top\rangle\}] \end{matrix}}}{\begin{matrix} 4x - 4z + 1 + 3 \leq 0 \\ [\{\langle y + 4x + n \leq 0, y + 4x + n = 0\rangle \mid \\ 0 \leq n < 3\} \cup \{\langle y + 4x + 2 \leq 0, \top\rangle\}] \end{matrix}} \quad \dfrac{\begin{matrix} -y - 4x - 1 \leq 0 & y + 4z - 2 \leq 0 \\ [\{\langle -y - 4x - 1 \leq 0, \top\rangle\}] & [\{\langle 0 \leq 0, \top\rangle\}] \end{matrix}}{\begin{matrix} -4x + 4z - 3 \leq 0 \\ [\{\langle -y - 4x - 1 \leq 0, \top\rangle\}] \end{matrix}}}{\begin{matrix} (1 \leq 0) \equiv \bot \\ [\{\langle n - 1 \leq 0, y + 4x + n = 0\rangle \mid 0 \leq n < 3\} \cup \{\langle 2 - 1 \leq 0, \top\rangle\}] \end{matrix}}$$

Drawback of Strengthen

- Interpolation of Strengthen creates potentially very big disjunctions
  - Linear in the strengthening factor $k := d\lceil \frac{c}{d} \rceil - c$
  - Can be exponential in the size of the proof

Example:
$$A := \begin{cases} -y - 4x - 1 \leq 0 \\ y + 4x \leq 0 \end{cases} \qquad B := \begin{cases} -y - 4z + 1 \leq 0 \\ y + 4z - 2 \leq 0 \end{cases}$$

Interpolant: $(y =_4 0) \vee (y + 1 =_4 0)$

Drawback of Strengthen

- Interpolation of Strengthen creates potentially very big disjunctions
  - Linear in the strengthening factor $k := d\lceil \frac{c}{d} \rceil - c$
  - Can be exponential in the size of the proof

Example:

$$A := \begin{cases} -y - 2nx - n + 1 \leq 0 \\ y + 2nx \leq 0 \end{cases} \quad B := \begin{cases} -y - 2nz + 1 \leq 0 \\ y + 2nz - n \leq 0 \end{cases}$$

Interpolant: $(y =_{2n} 0) \vee (y + 1 =_{2n} 0) \vee \ldots \vee (y =_{2n} n - 1)$

Drawback of Strengthen

- Interpolation of Strengthen creates potentially very big disjunctions
  - Linear in the strengthening factor $k := d\lceil \frac{c}{d} \rceil - c$
  - Can be exponential in the size of the proof

Example:

$$A := \begin{cases} -y - 2nx - n + 1 \leq 0 \\ y + 2nx \leq 0 \end{cases} \quad B := \begin{cases} -y - 2nz + 1 \leq 0 \\ y + 2nz - n \leq 0 \end{cases}$$

Interpolant: $(y =_{2n} 0) \vee (y + 1 =_{2n} 0) \vee \ldots \vee (y =_{2n} n - 1)$

- The problem are AB-mixed cuts:

$$\text{Strengthen} \; \frac{\sum_{x_i \notin B} c_i x_i + \sum_{y_j \notin A} c_j y_j + c \leq 0}{\sum_{x_i \notin B} c_i x_i + \sum_{y_j \notin A} c_j y_j + d \cdot \lceil \frac{c}{d} \rceil \leq 0}$$

Solution of [Kroening et al. LPAR'10]

- Avoid the problem by avoiding mixed cuts
  - Algorithm based on reduction to LA(Q) + Diophantine equations
- Generate interpolants linear in the size of proofs
- However, this is a strong restriction:
  - Forbids use of popular LA(Z) techniques like Gomory cuts, cuts from proofs [Dillig et al CAV'09], the Omega test [Pugh91]
  - Might generate much larger proofs

Solution of [Kroening et al. LPAR'10]

- Avoid the problem by avoiding mixed cuts
  - Algorithm based on reduction to LA(Q) + Diophantine equations
- Generate interpolants linear in the size of proofs
- However, this is a strong restriction:
  - Forbids use of popular LA(Z) techniques like Gomory cuts, cuts from proofs [Dillig et al CAV'09], the Omega test [Pugh91]
  - Might generate much larger proofs

Example:

$$A := \begin{cases} -y - 2nx - n + 1 \leq 0 \\ y + 2nx \leq 0 \end{cases} \quad B := \begin{cases} -y - 2nz + 1 \leq 0 \\ y + 2nz - n \leq 0 \end{cases}$$

Without AB-mixed cuts, proof of exponential size

Same interpolant as with Strengthen

In fact, [Kroening et al. LPAR'10] shows this is the only interpolant for (A, B)

Solution of [Kroening et al. LPAR'10]

FONDAZIONE BRUNO KESSLER

- Avoid the problem by avoiding mixed cuts
  - Algorithm based on reduction to LA(Q) + Diophantine equations
- Generate interpolants linear in the size of proofs
- However, this is a strong restriction:
  - Forbids use of popular LA(Z) techniques like Gomory cuts, cuts from proofs [Dillig et al CAV'09], the Omega test [Pugh91]
  - Might generate much larger proofs

Example:

$$A := \begin{cases} -y - 2nx \\ y + 2nx \leq \end{cases}$$

Implicit assumption: we are in the signature $\mathbb{Z} \cup \{+, \cdot, =, \leq\} \cup \{=_g\}_{g \in \mathbb{Z}^{>0}}$

Without AB-mixed cuts, proof of exponential size

Same interpolant as with Strengthen

In fact, [Kroening et al. LPAR'10] shows this is the only interpolant for (A, B)

Outline

- Background
- Current techniques for interpolation in LA(Z)
- A novel interpolation technique for LA(Z)
- Experimental evaluation

Interpolation with ceilings

- Idea: use a different extension of the signature of LA(Z), and extend also its domain
  - Introduce the ceiling function $\lceil \cdot \rceil$ [Pudlák '97]
  - Allow non-variable terms to be non-integers (e.g. $\frac{x}{2}$)
- Much simpler interpolation procedure
  - Proof annotations are single inequalities $(t \leq 0)$

Interpolation with ceilings

- Idea: use a different extension of the signature of LA(Z), and extend also its domain
  - Introduce the ceiling function $\lceil \cdot \rceil$ [Pudlák '97]
  - Allow non-variable terms to be non-integers (e.g. $\frac{x}{2}$)
- Much simpler interpolation procedure
  - Proof annotations are single inequalities $(t \leq 0)$

$$\text{Hyp}\ \frac{-}{t \leq 0\ [t' \leq 0]} \qquad \text{Comb}\ \frac{t_1 \leq 0\ [t_1' \leq 0] \qquad t_2 \leq 0\ [t_2' \leq 0]}{c_1 \cdot t_1 + c_2 \cdot t_2 \leq 0\ [c_1 \cdot t_1' + c_2 \cdot t_2' \leq 0]}$$

$$\text{Div}\ \frac{\begin{array}{c}\sum_{y_j \notin B} a_j y_j + \sum_{z_k \notin A} b_k z_k + \sum_{x_i \in A \cap B} c_i x_i + c \\ {[\sum_{y_j \notin B} a_j y_j + \sum_{x_i \in A \cap B} c_i' x_i + c']}\end{array}}{\begin{array}{c}\sum_{y_j \notin B} \frac{a_j}{d} y_j + \sum_{z_k \in B} \frac{b_k}{d} z_k + \sum_{x_i \in A \cap B} \frac{c_i}{d} x_i + \lceil \frac{c}{d} \rceil \\ {[\sum_{y_j \notin B} \frac{a_j}{d} y_j + \lceil \frac{\sum_{x_i \in A \cap B} c_i' x_i + c'}{d} \rceil]}\end{array}} \quad d > 0 \text{ divides } a_j, b_k, c_i$$

Interpolation with ceilings - example

- No blowup of interpolants wrt. the size of the proofs

$$A := \begin{cases} -y - 2nx - n + 1 \leq 0 \\ y + 2nx \leq 0 \end{cases} \qquad B := \begin{cases} -y - 2nz + 1 \leq 0 \\ y + 2nz - n \leq 0 \end{cases}$$

$$\dfrac{\dfrac{\dfrac{y + 2nx \leq 0 \qquad -y - 2nz + 1 \leq 0}{2nx - 2nz + 1 \leq 0}}{2n \cdot (x - z + 1 \leq 0)} \qquad \dfrac{-y - 2nx - n + 1 \leq 0 \qquad y + 2nz - n \leq 0}{-2nx + 2nz - 2n + 1 \leq 0}}{(1 \leq 0) \equiv \bot}$$

Interpolation with ceilings - example

- No blowup of interpolants wrt. the size of the proofs

$$A := \begin{cases} -y-2nx-n+1 \le 0 \\ y+2nx \le 0 \end{cases} \qquad B := \begin{cases} -y-2nz+1 \le 0 \\ y+2nz-n \le 0 \end{cases}$$

$$\dfrac{\dfrac{\dfrac{\begin{array}{cc} y+2nx \le 0 & -y-2nz+1 \le 0 \\ [y+2nx \le 0] & [0 \le 0] \end{array}}{\begin{array}{c} 2nx-2nz+1 \le 0 \\ [y+2nx \le 0] \end{array}}}{\begin{array}{c} 2n \cdot (x-z+1 \le 0) \\ [x+\lceil \frac{y}{2n} \rceil \le 0] \end{array}} \qquad \dfrac{\begin{array}{cc} -y-2nx-n+1 \le 0 & y+2nz-n \le 0 \\ [-y-2nx-n+1 \le 0] & [0 \le 0] \end{array}}{\begin{array}{c} -2nx+2nz-2n+1 \le 0 \\ [-y-2nx-n+1 \le 0] \end{array}}}{\begin{array}{c} (1 \le 0) \equiv \bot \\ [2n\lceil \frac{y}{2n} \rceil - y - n + 1 \le 0] \end{array}}$$

Interpolation with ceilings - example

- No blowup of interpolants wrt. the size of the proofs

$$A := \begin{cases} -y - 2nx - n + 1 \leq 0 \\ y + 2nx \leq 0 \end{cases} \qquad B := \begin{cases} -y - 2nz + 1 \leq 0 \\ y + 2nz - n \leq 0 \end{cases}$$

Interpolant:
$(2n\lceil \frac{y}{2n} \rceil - y - n + 1 \leq 0)$

$$\dfrac{\dfrac{\dfrac{\begin{matrix} y + 2nx \leq 0 & -y - 2nz + 1 \leq 0 \\ [y + 2nx \leq 0] & [0 \leq 0] \end{matrix}}{\begin{matrix} 2nx - 2nz + 1 \leq 0 \\ [y + 2nx \leq 0] \end{matrix}}}{\begin{matrix} 2n \cdot (x - z + 1 \leq 0) \\ [x + \lceil \frac{y}{2n} \rceil \leq 0] \end{matrix}} \qquad \dfrac{\begin{matrix} -y - 2nx - n + 1 \leq 0 & y + 2nz - n \leq 0 \\ [-y - 2nx - n + 1 \leq 0] & [0 \leq 0] \end{matrix}}{\begin{matrix} -2nx + 2nz - 2n + 1 \leq 0 \\ [-y - 2nx - n + 1 \leq 0] \end{matrix}}}{\begin{matrix} (1 \leq 0) \equiv \bot \\ [2n\lceil \frac{y}{2n} \rceil - y - n + 1 \leq 0] \end{matrix}}$$

SMT(LA(Z)) with ceilings

- Like modular equations, also ceilings can be eliminated via preprocessing:
  - Replace every term $\lceil t \rceil$

    with a fresh integer variable $x_{\lceil t \rceil}$
  - Add the 2 unit clauses

    (encoding the meaning of ceiling: $\lceil t \rceil - 1 < t \leq \lceil t \rceil$)

    $$(l \cdot x_{\lceil t \rceil} - l \cdot t + l \leq 0)$$

    $$(l \cdot t - l \cdot x_{\lceil t \rceil} \leq 0)$$

    where l is the least common multiple of the denominators of the coefficients in t

Outline

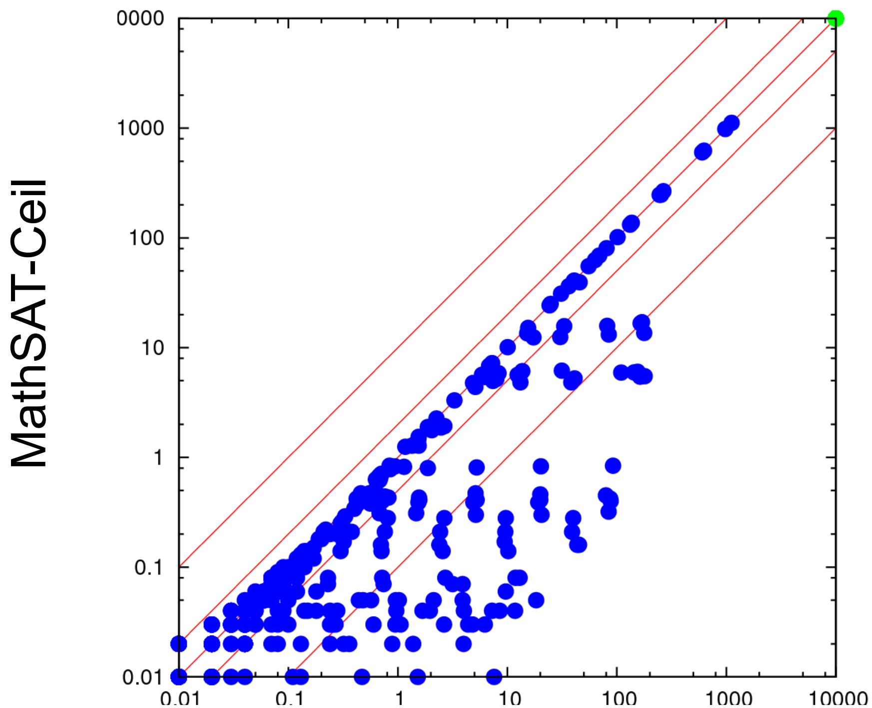
- Background
- Current techniques for interpolation in LA(Z)
- A novel interpolation technique for LA(Z)
- Experimental evaluation

Experiments

- Implementation on top of MathSAT 5
  - Use also algorithm for Diophantine equations and Boolean interpolation algorithm for dealing with Branch and Bound
  - Implemented both algorithm based on Strengthen (MathSAT-ModEq) and on ceilings (MathSAT-Ceil)
- Use benchmarks that require non-trivial integer reasoning

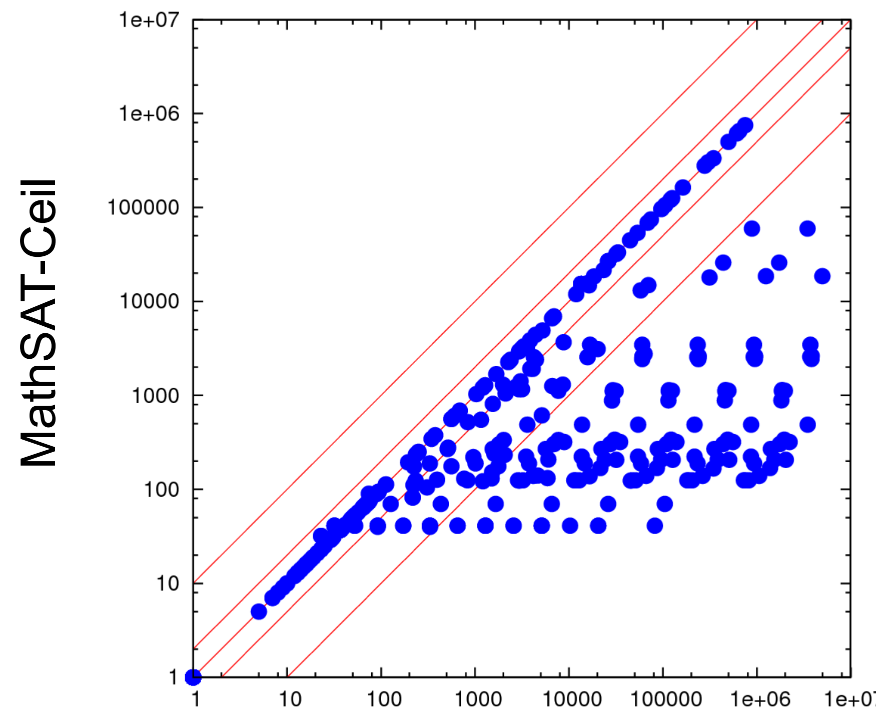
Results – Strengthen vs. ceilings

Execution Time



MathSAT-ModEq

Size of Interpolants



MathSAT-ModEq

Thank You