

Formalization of a Proof Calculus for Incremental Linearization for NTA Satisfiability

Tomaz Mascarenhas¹, Harun Khan¹, Abdalrhman Mohamed, Andrew Reynolds, Haniel Barbosa, Clark Barrett, Cesare Tinelli



Belo Horizonte, Brasil, 11/04/2025

¹Equal contribution

Aritmética Não-linear e Funções Transcendentais

- ▷ A teoria NTA em SMT compreende aritmética não-linear e algumas funções transcendentais
 - ▶ Funções trigonométricas, exp e log.
- ▷ Aplicações em: verificação formal de sistemas dinâmicos, processamento de sinais digitais e planejamento de movimentação para robôs.
 - ▶ Automatização em assistentes de demonstração, se for provido o ferramental necessário
- ▷ Complexidade $\mathcal{O}(2^{2^n})$ para aritmética não-linear em \mathbb{R} e indecidível para NTA.

Incremental Linearization

- ▷ *Incremental linearization* é um método para decidir a satisfatibilidade de fórmulas em NTA.
 - ▶ Não é um procedimento de decisão
- ▷ Dada uma fórmula ψ em NTA, abstraem-se todas as multiplicações e funções transcendentais usando funções não-interpretadas, obtendo-se uma fórmula ψ' em UFLRA.
- ▷ Se ψ' não é satisfatível, então ψ também não é.

Incremental Linearization

- ▷ Se ψ' é satisfatível, não necessariamente ψ também é. Nesse caso, o método introduz incrementalmente novos lemas a ψ' bloqueando soluções espúrias.

Exemplo

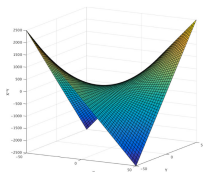
Seja $\psi = x^2 + y^2 \leq 2 \wedge (x \leq -1.1 \vee x \geq 1.1) \wedge (y \leq -1.1 \vee y \geq 1.1)$.

ψ' seria o mesmo, mas em vez de x^2 e y^2 teríamos $f_*(x, x)$ e $f_*(y, y)$.

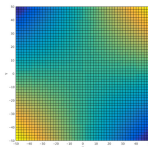
É possível interpretar f_* tal que ψ' seja satisfeita, mas a fórmula original não é satisfatível.

Incremental Linearization

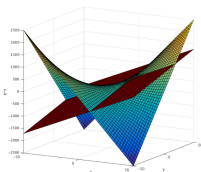
- ▶ Soluções espúrias envolvendo multiplicações de variáveis são bloqueadas usando a *equação do plano tangente* e o sinal das variáveis.
- ▶ Soluções espúrias envolvendo funções transcendentais são bloqueadas usando o *polinômio de Taylor*.
 - ▶ É a ferramenta perfeita para aproximar incrementalmente uma função derivável



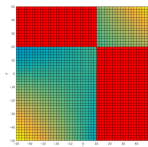
(a) $x * y$



(b) $x * y$ (top view)



(c) $x * y$ and tangent plane



(d) $x * y$ and tangent plane (top view)

Incremental Linearization

```
1: function INCREMENTALLINEARIZATION( $\psi$ )
2:    $\psi' \leftarrow \text{INITIALABSTRACTION}(\psi)$ 
3:    $\Gamma \leftarrow \emptyset$ 
4:   while INTIMELIMIT() do
5:      $\langle sat, \mu \rangle \leftarrow \text{SOLVEUFLRA}(\psi' \wedge \Gamma)$ 
6:     if not  $sat$  then return UNSAT
7:      $\langle sat', \Gamma' \rangle \leftarrow \text{CHECKREFINE}(\psi', \mu)$ 
8:     if  $sat'$  then return SAT
9:      $\Gamma \leftarrow \Gamma \cup \Gamma'$ 
```

- ▷ cvc5 implementa uma versão modificada desse algoritmo, apropriada para suas otimizações internas.
- ▷ O solucionador define um *proof calculus* composto por 21 lemas que capturam raciocínio por trás do algoritmo, possibilitando a produção de demonstrações.
- ▷ Esses lemas são, em sua grande maioria, as regras usadas para eliminar soluções espúrias.

- ▷ Neste trabalho nós mecanizamos demonstrações de todas os lemas presentes no proof calculus, fortalecendo a confiança no algoritmo
- ▷ Nesse processo, identificamos algumas hipóteses faltando na documentação de algumas regras, além de outras imprecisões.
- ▷ Além disso, é um primeiro passo em direção a reconstrução de demonstrações em Lean.

Lemas Formalizados

Lemas para multiplicação - ARITH_MULT_TANGENT

$$\frac{- \mid x, y, a, b}{xy \leq bx + ay - ab \leftrightarrow ((x \leq a \wedge y \geq b) \vee (x \geq a \wedge y \leq b))}$$

$$\frac{- \mid x, y, a, b}{xy \geq bx + ay - ab \leftrightarrow ((x \leq a \wedge y \leq b) \vee (x \geq a \wedge y \geq b))}$$

Lemas para multiplicação - ARITH_MULT_TANGENT

theorem arithMulTangentLower $\{\alpha : \text{Type}\}$ [LinearOrderedField α] (x y a b : α) :
 $x * y \leq b * x + a * y - a * b \leftrightarrow ((x \leq a \wedge y \geq b) \vee (x \geq a \wedge y \leq b))$

theorem arithMulTangentUpper $\{\alpha : \text{Type}\}$ [LinearOrderedField α] (x y a b : α) :
 $x * y \geq b * x + a * y - a * b \leftrightarrow ((x \leq a \wedge y \leq b) \vee (x \geq a \wedge y \geq b))$

Lemas para multiplicação - ARITH_MULT_SIGN

$$\frac{- \mid f_1, \dots, f_k, m}{(f_1 \wedge \dots \wedge f_k) \rightarrow m \bowtie 0}$$

Cada f_i é da forma $x_i \bowtie 0$. m é um monômio composto por potências dos x_i 's. A conclusão é definida com base na paridade dos expoentes e no sinal de cada x_i .

Lemas para multiplicação - ARITH_MULT_SIGN

```
theorem powNegOdd      :  $\forall \{k : \mathbb{N}\} \{r : \mathbb{R}\}, r < 0 \rightarrow \text{Odd } k \rightarrow r ^ k < 0$   
theorem powNegEven    :  $\forall \{k : \mathbb{N}\} \{r : \mathbb{R}\}, r < 0 \rightarrow \text{Even } k \rightarrow r ^ k > 0$   
theorem nonZeroEvenPow :  $\forall \{k : \mathbb{N}\} \{r : \mathbb{R}\}, r \neq 0 \rightarrow \text{Even } k \rightarrow r ^ k > 0$   
theorem powPos        :  $\forall \{k : \mathbb{N}\} \{r : \mathbb{R}\}, r > 0 \rightarrow r ^ k > 0$ 
```

```
theorem combineSigns1 :  $\forall \{a \ b : \mathbb{R}\}, a > 0 \rightarrow b > 0 \rightarrow b * a > 0$   
theorem combineSigns2 :  $\forall \{a \ b : \mathbb{R}\}, a > 0 \rightarrow b < 0 \rightarrow b * a < 0$   
theorem combineSigns3 :  $\forall \{a \ b : \mathbb{R}\}, a < 0 \rightarrow b > 0 \rightarrow b * a < 0$   
theorem combineSigns4 :  $\forall \{a \ b : \mathbb{R}\}, a < 0 \rightarrow b < 0 \rightarrow b * a > 0$ 
```

Lemas para funções transcendentais - Limitando \exp e \sin

$$\frac{- \mid d, c, t}{t \geq c \rightarrow \exp(t) \geq \mathbf{maclaurin}(\exp, d, c)}$$

Onde d é um número ímpar.

$$\frac{- \mid d, t, l, u}{(t \geq l \wedge t \leq u) \rightarrow \sin(t) \geq \mathbf{secant}(\sin, l, u, t)}$$

Onde d é congruente a 3 módulo 4*, \sin é côncavo no intervalo $[l, u]^*$ e \mathbf{secant} é a reta secante ao d -ésimo polinômio de taylor nos pontos $(l, p(l))$ e $(r, p(r))$.

Termo complementar de Lagrange

- ▷ Esses resultados seguem do *Teorema de Taylor com termo complementar de Lagrange*:

$$f(x) - \left(\sum_{j=0}^n \frac{f^{(j)}(x_0)}{j!} (x - x_0)^j \right) = \frac{f^{(n+1)}(x')}{(n+1)!} (x - x_0)^{n+1}$$

- ▷ Não seria possível formalizar esses resultados sem a Mathlib. O resultado acima existe nela, mas assumindo que $x_0 < x$. Nós estendemos a biblioteca com a versão geral do teorema.

Lemas para funções transcendentais - Limitando \exp e \sin

```
theorem arithTransExpApproxBelow (d n :  $\mathbb{N}$ ) ( $\_ : d = 2*n + 1$ ) :  
  Real.exp x  $\geq$  taylorWithinEval Real.exp d Set.univ 0 x
```

```
theorem arithTransSineApproxBelowPos (d k :  $\mathbb{N}$ ) (hd :  $d = 4 * k + 3$ ) (t l u :  $\mathbb{R}$ )  
  (ht :  $l \leq t \wedge t \leq u$ )  
  (hl :  $0 < l$ ) (hu :  $u \leq \text{Real.pi}$ ) :  
  let p :  $\mathbb{R} \rightarrow \mathbb{R} := \text{taylorWithinEval Real.sin d Set.univ 0}$   
  sin t  $\geq ((p\ l - p\ u) / (l - u)) * (t - l) + p\ l$ 
```


Outras funções transcendentais

$$\triangleright \cos(x) = \sin(x + \frac{\pi}{2})$$

$$\triangleright \tan(x) = \frac{\sin(x)}{\cos(x)}$$

$$\triangleright \sec(x) = \frac{1}{\cos(x)}$$

$$\triangleright \csc(x) = \frac{1}{\sin(x)}$$

$$\triangleright \cot(x) = \frac{1}{\tan(x)}$$

\triangleright log e trigonométricas inversas são modeladas usando novas variáveis e asserções. Por exemplo, para um termo com a forma $\log(1 + x)$ introduzimos uma variável y e adicionamos a asserção $\exp(y) = 1 + x$.

- ▷ Correções no cvc5
- ▷ Reconstrução de demonstrações usando *lean-smt*
- ▷ Desafios com números reais