Formalization of a Proof Calculus for Incremental Linearization for NTA Satisfiability

Tomaz Mascarenhas[1], Harun Khan[1], Abdalrhman Mohamed, Andrew Reynolds, Haniel Barbosa, Clark Barrett, Cesare Tinelli

UFMG | ICEx
DEPARTAMENTO DE
CIÊNCIA DA COMPUTAÇÃO

UFMG

Belo Horizonte, Brasil, 11/04/2025

[1]Equal 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

Incremental Linearization

function $\textsc{IncrementalLinearization}(\psi)$
    $\psi' \leftarrow \textsc{InitialAbstraction}(\psi)$
    $\Gamma \leftarrow \varnothing$
    while $\textsc{InTimeLimit}()$ **do**
        $\langle sat, \mu \rangle \leftarrow \textsc{SolveUFLRA}(\psi' \wedge \Gamma)$
        if not sat **then return** UNSAT
        $\langle sat', \Gamma' \rangle \leftarrow \textsc{CheckRefine}(\psi', \mu)$
        if sat' **then return** SAT
        $\Gamma \leftarrow \Gamma \cup \Gamma'$

Proof Calculus

- 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.

Contribuições

- 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

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

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

Lemas para multiplicação - ARITH_MULT_TANGENT

```
theorem arithMulTangentLower {α : Type} [LinearOrderedField α] (x y a b : α) :
    x * y ≤ b * x + a * y - a * b ↔ ((x ≤ a ∧ y ≥ b) ∨ (x ≥ a ∧ y ≤ b))
```

```
theorem arithMulTangentUpper {α : Type} [LinearOrderedField α] (x y a b : α) :
    x * y ≥ b * x + a * y - a * b ↔ ((x ≤ a ∧ y ≤ b) ∨ (x ≥ a ∧ y ≥ b))
```

Lemas para multiplicação - ARITH_MULT_SIGN

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

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      : ∀ {k : ℕ} {r : ℝ}, r < 0 → Odd k  → r ^ k < 0
theorem powNegEven     : ∀ {k : ℕ} {r : ℝ}, r < 0 → Even k → r ^ k > 0
theorem nonZeroEvenPow : ∀ {k : ℕ} {r : ℝ}, r ≠ 0 → Even k → r ^ k > 0
theorem powPos         : ∀ {k : ℕ} {r : ℝ}, r > 0          → r ^ k > 0
```

```
theorem combineSigns₁ : ∀ {a b : ℝ}, a > 0 → b > 0 → b * a > 0
theorem combineSigns₂ : ∀ {a b : ℝ}, a > 0 → b < 0 → b * a < 0
theorem combineSigns₃ : ∀ {a b : ℝ}, a < 0 → b > 0 → b * a < 0
theorem combineSigns₄ : ∀ {a b : ℝ}, a < 0 → b < 0 → b * a > 0
```

Lemas para funções transcendentais - Limitando $\exp$ e $\sin$

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

Onde d é um número ímpar.

$$\frac{- \mid d, t, l, u}{(t \geq l \land t \leq u) \to \sin(t) \geq \texttt{secant}(\sin, l, u, t)}$$

Onde d é congruente a 3 módulo 4^*, $\sin$ é côncavo no intervalo $[l, u]^*$ e `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 : ℕ) (_ : d = 2*n + 1) :
  Real.exp x ≥ taylorWithinEval Real.exp d Set.univ 0 x
```

```
theorem arithTransSineApproxBelowPos (d k : ℕ) (hd : d = 4 * k + 3) (t l u : ℝ)
    (ht : l ≤ t ∧ t ≤ u)
    (hl : 0 < l) (hu : u ≤ Real.pi) :
    let p : ℝ → ℝ := taylorWithinEval Real.sin d Set.univ 0
    sin t ≥ ((p l - p u) / (l - u)) * (t - l) + p l
```

Outras funções transcendentais

- $\cos(x) = \sin(x + \frac{\pi}{2})$
- $\tan(x) = \frac{\sin(x)}{\cos(x)}$
- $\sec(x) = \frac{1}{\cos(x)}$
- $\csc(x) = \frac{1}{\sin(x)}$
- $\cot(x) = \frac{1}{\tan(x)}$
- $\log$ 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 $\exp(y) = 1 + x$.

Trabalhos futuros

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