On Non-Looping Term Rewriting & **Proving Non-Looping Non-Termination Automatically**

Astrid van der Jagt

In this presentation

Non-Looping Non-Termination of TRS

- Term Rewrite Systems (TRS)
- Paper 1: On Non-Looping Term Rewriting
 - Loops/Non-Loops
 - Inner-looping TRS
 - Normal TRS
- Paper 2: Proving Non-Looping Non-Termination Automatically
 - Pattern terms and pattern rules
 - Proving Non-Termination

Term Rewrite Systems (TRS) (1/7)

- Terms
 - Signature F
 - Infinite set of variables V
- Rules

function symbols + arity

- Context
- Substitution

Term Rewrite Systems (TRS) (2/7)

Terms and Rules

- Terms:
 - *F* = { plus (2), s (1), 0 (0) } *x*, *y* ∈ *V*
- Rules:
 - plus(\mathbf{X} , 0) → \mathbf{X}
 - $\circ \quad \mathsf{plus}(\mathbf{x},\, \mathsf{s}(\mathbf{y})) \qquad \rightarrow \mathsf{plus}(\mathsf{s}(\mathbf{x}),\, \mathbf{y})$
- $s \rightarrow_{R} t$ if there exists a rewrite rule $I \rightarrow r$, a substitution σ and a context *C* such that $s = C[I\sigma]$ and $t = C[r\sigma]$

Term Rewrite Systems (TRS) (3/7)

Context

• $s \rightarrow_{R} t$ if there exists a rewrite rule $I \rightarrow r$, a substitution σ and a context *C* such that $s = C[I\sigma]$ and $t = C[r\sigma]$

Let C be a context with a hole □
Then C[t] is the term obtained from replacing □ with a term t.

• Example:

$$C = f(x, g(\Box, z)) C[t] = f(x, g(t, z))$$

Term Rewrite Systems (TRS) (4/7)

Substitution

• $s \rightarrow_{R} t$ if there exists a rewrite rule $I \rightarrow r$, a substitution σ and a context *C* such that $s = C[I\sigma]$ and $t = C[r\sigma]$

• A substitution is a mapping σ from V to the Terms.

 $\sigma = [\mathbf{x} \mapsto s(s(0)), \, \mathbf{y} \mapsto s(0)]$

• Applying σ to a term t = plus(**x**, s(**y**)), we get

 $t \sigma = plus(s(s(0)), s(s(0)))$

Term Rewrite Systems (TRS) (5/7)

Rewriting

- $s \rightarrow_{R} t$ if there exists a rewrite rule $I \rightarrow r$, a substitution σ and a context *C* such that $s = C[I\sigma]$ and $t = C[r\sigma]$
- Rules:
 - \circ plus(x, 0) → x
 - $\circ \quad \mathsf{plus}(\mathbf{x},\, \mathsf{s}(\mathbf{y})) \qquad \rightarrow \mathsf{plus}(\mathsf{s}(\mathbf{x}),\, \mathbf{y})$

C = 🗆

 $\sigma = [\mathbf{x} \mapsto \mathbf{s}(\mathbf{s}(0)), \, \mathbf{y} \mapsto \mathbf{s}(0)]$

• $plus(s(s(0)), s(s(0))) \rightarrow_R plus(s(s(s(0))), s(0))$ $\rightarrow_R plus(s(s(s(s(0)))), 0))$ $\rightarrow_R s(s(s(s(0))))$

Term Rewrite Systems (TRS) (6/7)

Non-Termination

- Rules:
 - \circ plus(x, 0) → x
 - $\circ \quad \mathsf{plus}(\mathbf{x},\,\mathsf{s}(\mathbf{y})) \qquad \rightarrow \mathsf{plus}(\mathbf{y},\,\mathsf{s}(\mathbf{x}))$

• $plus(s(s(0)), s(s(0))) \rightarrow_R plus(s(0), s(s(s(0)))) \rightarrow_R plus(s(s(0)), s(s(0))) \rightarrow_R \dots$

Term Rewrite Systems (TRS) (7/7)

Termination

• A TRS is terminating if it does not admit any infinite reduction sequences

On Non-Looping Term Rewriting

Yi Wang and Masahiko Sakai (2006)

Loops Looping TRS

 $\rightarrow_{\mathsf{R}} \cdots$

A reduction sequence loops if it contains $t \rightarrow_R^+ C[t\sigma]$. A TRS admits a loop if there is a looping reduction sequence.

$$\begin{split} f(x) & \rightarrow h(f(g(x))) & t = f(x) \\ & C = h(\Box) \\ f(x) & \rightarrow_R h(f(g(x))) & \sigma = [x \mapsto g(x)] \\ & \rightarrow_R h(h(f(g(g(g(x)))))) & \rightarrow_R h(h(h(f(g(g(g(x)))))))) \end{split}$$

Non-Looping TRS

A rewrite sequence is non-looping if it is infinite and does not contain any loop. A TRS is non-looping if it admits an non-looping sequence. A TRS is properly non-looping if it is non-looping and does not admit any looping sequence.

 $\begin{array}{ll} b(c) & \rightarrow d(c) \\ b(d(x)) & \rightarrow d(b(x)) \\ a(d(x)) & \rightarrow a(b(b(x))) \end{array}$

a(b(c)) →_R² a(b(b(c))) →_R ³ a(b(b(b(c)))) →_R ⁴ ...

Non-Loops (2/5)

Inner-Looping sequence/TRS

Given a TRS R, let t be a term, an *inner-looping sequence* is of the form:

 $C[\Delta^{\ell_1} t \overline{\delta}^{\ell_1}] \longrightarrow_{\mathsf{R}}^{+} C[\Delta^{\ell_2} t \overline{\delta}^{\ell_2}] \longrightarrow_{\mathsf{R}}^{+} \dots$

Where *C* and Δ are contexts, δ is a substitution, $\{\ell_i\}$ is an infinite sequence of natural numbers.

A TRS R is *inner-looping* if R admits an inner-looping sequence. A TRS R is *properly inner-looping* if R is inner-looping and does not admit any looping sequence.



 $C[\Delta^{\ell_1} t \delta^{\ell_1}] \longrightarrow_R^+ C[\Delta^{\ell_2} t \delta^{\ell_2}] \longrightarrow_R^+ \dots$ for term t, where *C* and Δ are contexts, δ is a substitution, $\{\ell_i\}$ is an infinite sequence.

$$\begin{array}{lll} b(c) & \rightarrow d(c) & a(b(c)) & \rightarrow_{R}^{2} a(b(b(c))) \\ b(d(x)) & \rightarrow d(b(x)) & & \\ a(d(x)) & \rightarrow a(b(b(x))) & & & \rightarrow_{R}^{3} a(b(b(b(c)))) \\ & & \rightarrow_{R}^{4} \dots \end{array}$$

t = c, C = a(
$$\Box$$
), Δ = b(\Box), δ = Ø, ℓ_i = i



Are there non-looping rewrite sequences without any patterns?

Normal numbers: real numbers whose digits show a random distribution with all digits appearing equally.

Normal sequence: infinite reduction sequence with all function symbols appearing equally in every term of the sequence.

Normal TRS is a TRS that admits a normal sequence.



The inner-looping property and the properly inner-looping property for TRSs are undecidable.

The non-looping property is undecidable.

Conclusion

On Non-Looping Term Rewriting

- Looping TRS & Non-looping TRS
- Inner-looping TRS
- Normal TRS
- Undecidability

Proving Non-Looping Non-Termination Automatically

Fabian Emmes, Tim Enger and Jürgen Giesl (2012)

Proving Non-Looping Non-Termination Automatically

- Pattern terms
- Pattern rules
- The technique: narrowing

Our example (1/2)

Proving non-looping non-termination automatically

def f(): while (gt(x, y)): x = dbl(x)y = y + 1

Our example (2/2)

Proving non-looping non-termination automatically

f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y)) gt(s(x), 0) \rightarrow tt **gt**(0, y) \rightarrow ff gt(s(x), s(y)) \rightarrow gt(x, y) \rightarrow times(s(s(0)), x) dbl(x)times(x, 0) $\rightarrow 0$ times(x, s(y)) \rightarrow plus(times(x, y), x) plus(x, 0) $\rightarrow X$ $plus(x, s(y)) \rightarrow plus(s(x), y)$

def f():

Х

V

while (gt(x, y)):

 \equiv

=

dbl(x)

v + 1

Pattern Terms (1/2)

Describing a set of terms

- A mapping $n \mapsto t \sigma^n \mu$
 - t in Terms
 - σ , μ are substitutions

$$\begin{array}{ll} gt(s(\textbf{x}),\,0) & \longrightarrow tt\\ gt(s(\textbf{x}),\,s(\textbf{y})) & \longrightarrow gt(\textbf{x},\,\textbf{y}) \end{array}$$

Base termPumping substitutionClosing substitutiongt(s(x), s(y)) $[x \mapsto s(x), y \mapsto s(y)]^n$ $[x \mapsto s(x), y \mapsto 0]$

Pattern Terms (2/2)

Interpreting a pattern term

A mapping $n \mapsto t \sigma^n \mu$

For n = 1, we apply the pumping substitution once to the base term, and then apply the closing substitution: gt(s(s(x)), s(s(y))) gt(s(s(s(x))), s(s(0)))

Base termPumping substitutionClosing substitutiongt(s(x), s(y)) $[x \mapsto s(x), y \mapsto s(y)]^n$ $[x \mapsto s(x), y \mapsto 0]$

Pattern Rules

Describing a set of rewrite sequences

 $\begin{array}{ll} gt(s(\textbf{x}),\,0) & \longrightarrow tt \\ gt(s(\textbf{x}),\,s(\textbf{y})) & \longrightarrow gt(\textbf{x},\,\textbf{y}) \end{array}$

A pattern rule $p \hookrightarrow q$ is correct w.r.t. a TRS R if $p(n) \rightarrow_R^+ q(n)$ for all $n \in \mathbb{N}$. gt(sⁿ⁺¹(x), sⁿ⁺¹(y)) gt(sⁿ⁺²(x), sⁿ⁺¹(0)) and it holds that gt(sⁿ⁺²(x), sⁿ⁺¹(0)) \rightarrow_R^+ tt

Base term Base term gt(s(x), s(y)) gt(s(x), s(y))	Pumping substitution Pumping substitution $[x \mapsto s(x), y \mapsto s(y)]^n$ $[x \mapsto s(x), y \mapsto s(y)]^n$	Closing substitution Closing substitution $[x \mapsto s(x), y \mapsto 0]$ $[x \mapsto s(x), y \mapsto 0]$
tt	∽ ⊘n	Ø
tt	⊘n	Ø

Proving Non-Looping Non-Termination Automatically

The technique

- 1. Generate pattern rules from the TRS.
- 2. Modify to avoid empty pattern substitutions.
- 3. Prepare for narrowing $p \rightarrow q$ with $p' \rightarrow q'$:
 - a. Make the base term of q at some position equal to the base term of p'.
 - b. Make the pumping substitutions and closing substitutions of the two pattern rules equal.
- 4. Narrow $p \hookrightarrow q$ with $p' \hookrightarrow q'$.
- 5. Check for non-termination of the resulting rule $s \rightarrow t$: is t a specialization of s?

Detecting Non-Termination

$$f(tt, \mathbf{x}, \mathbf{y}) \rightarrow f(gt(\mathbf{x}, \mathbf{y}), dbl(\mathbf{x}), s(\mathbf{y}))$$

Let $s \sigma^n \mu \hookrightarrow t (\sigma^m \theta)^n (\mu v)$ be correct w.r.t. a TRS R, where θ commutes with both σ and μ . If there is a position π and some $b \in \mathbb{N}$ such that $s \sigma^b = t|_{\pi}$, then R is non-terminating.

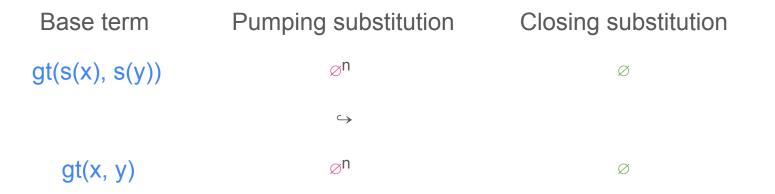
 $\mathbf{s} \sigma^n \mu \rightarrow_R^+ \mathbf{t} (\sigma^m \theta)^n (\mu v)$ and if we zoom in on the subterm where $\mathbf{s} \sigma^b = \mathbf{t}|_{\pi}$ we get

 $t|_{\pi} (\sigma^{m} \theta)^{n} (\mu v) = s \sigma^{b} (\sigma^{m} \theta)^{n} (\mu v)$ $= s \sigma^{mn+b} \mu \theta^{n} v$

Creating Pattern Rules (1/16)

Using *inference rules*

 $s \otimes^n \otimes \hookrightarrow t \otimes^n \otimes if s \to t \in R$



 $gt(s(\mathbf{x}), s(\mathbf{y}))$

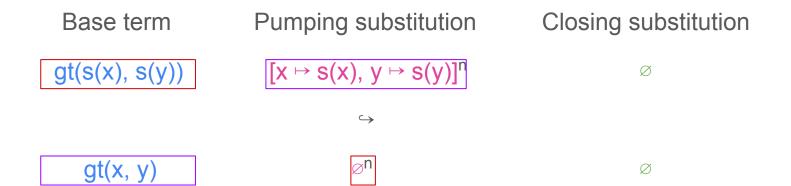
 \rightarrow gt(x, y)

Creating Pattern Rules (2/16)

Using *inference rules*

 $s \otimes^n \otimes \hookrightarrow t \otimes^n \otimes$ if $s\theta = t\sigma$, and

s $\sigma^n \oslash \hookrightarrow t \Theta^n \oslash \Theta$ commutes with σ



 $gt(s(\mathbf{x}), s(\mathbf{y}))$

 \rightarrow gt(x, y)

Creating Pattern Rules (3/16)

Equivalence by *domain renaming*

- $p \hookrightarrow q$ if p is equivalent to p' and
- $p' \hookrightarrow q'$ q is equivalent to q'

Base termPumping substitutionClosing substitutiongt(s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto x, y' \mapsto y]$ \hookrightarrow \bigotimes

 $gt(s(\mathbf{x}), s(\mathbf{y}))$

 \rightarrow gt(x, y)

Creating Pattern Rules (4/16)

$$gt(s(\mathbf{x}), s(\mathbf{y})) \longrightarrow gt(\mathbf{x}, \mathbf{y})$$

Equivalence by irrelevant pattern substitutions

- $p \hookrightarrow q$ if p is equivalent to p' and
- $p' \hookrightarrow q'$ q is equivalent to q'

Base termPumping substitutionClosing substitutiongt(s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto x, y' \mapsto y]$ \hookrightarrow gt(x, y) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto x, y' \mapsto y]$

Creating Pattern Rules (5/16) Instantiation

 $\begin{array}{cc} gt(s(\mathbf{x}), 0) & \longrightarrow tt \\ gt(s(\mathbf{x}), s(\mathbf{y})) & \longrightarrow gt(\mathbf{x}, \mathbf{y}) \end{array}$

We instantiate x to be s(x) and y to be 0.

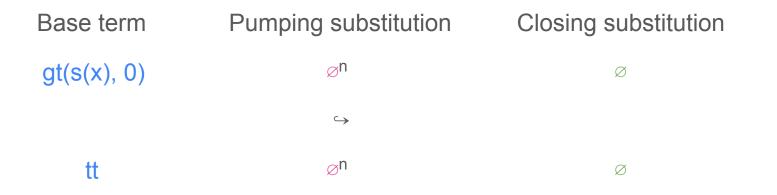
Base termPumping substitutionClosing substitutiongt(s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow gt(s(x), 0) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$

Creating Pattern Rules (6/16)

Pattern rule from TRS

 $\begin{array}{ll} gt(s(\textbf{x}),\,0) & \rightarrow tt \\ gt(s(\textbf{x}),\,s(\textbf{y})) & \rightarrow gt(\textbf{x},\,\textbf{y}) \end{array}$

 $\mathbf{s} \otimes^{\mathsf{n}} \otimes \hookrightarrow \mathbf{t} \otimes^{\mathsf{n}} \otimes \quad \text{if } \mathbf{s} \to \mathbf{t} \in \mathbf{R}$



Creating Pattern Rules (7/16)

Equivalence by irrelevant pattern substitutions

 $p \hookrightarrow q$ if p is equivalent to p' and

 $p' \hookrightarrow q'$ q is equivalent to q'

Since x' and y' are not relevant in the pattern rule:

Base termPumping substitutionClosing substitutiongt(s(x), 0) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$

 $\begin{array}{ll} gt(s(\textbf{x}),\,0) & \longrightarrow tt\\ gt(s(\textbf{x}),\,s(\textbf{y})) & \longrightarrow gt(\textbf{x},\,\textbf{y}) \end{array}$

Creating Pattern Rules (8/16)

Narrowing

 $\begin{array}{ll} \operatorname{gt}(\operatorname{s}(\mathbf{x}),\, 0) & \longrightarrow \operatorname{tt} \\ \operatorname{gt}(\operatorname{s}(\mathbf{x}),\,\operatorname{s}(\mathbf{y})) & \longrightarrow \operatorname{gt}(\mathbf{x},\,\mathbf{y}) \end{array}$

Base term Pumping substitution Closing substitution $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$ gt(s(x'), s(y')) \hookrightarrow $[\mathbf{X}' \mapsto \mathbf{S}(\mathbf{X}'), \mathbf{Y}' \mapsto \mathbf{S}(\mathbf{Y}')]^n [\mathbf{X}' \mapsto \mathbf{S}(\mathbf{X}), \mathbf{Y}' \mapsto \mathbf{0}]$ gt(s(x), 0)Base term Pumping substitution Closing substitution gt(s(x), 0) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow tt $[\mathbf{X}' \mapsto \mathbf{S}(\mathbf{X}'), \mathbf{Y}' \mapsto \mathbf{S}(\mathbf{Y}')]^n [\mathbf{X}' \mapsto \mathbf{S}(\mathbf{X}), \mathbf{Y}' \mapsto \mathbf{0}]$

Creating Pattern Rules (9/16) Narrowing

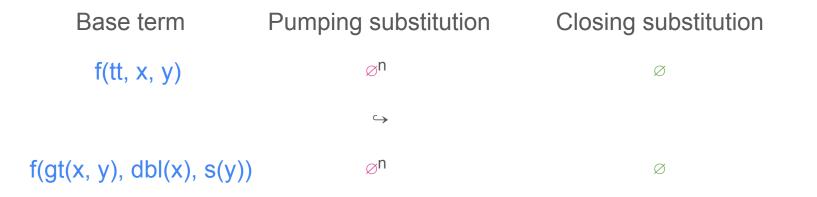
Base termPumping substitutionClosing substitutiongt(s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow $[x' \mapsto s(x'), y' \mapsto s(y')]^n [x' \mapsto s(x), y' \mapsto 0]$

Creating Pattern Rules (10/16)

Pattern rule from TRS

 $f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y))$

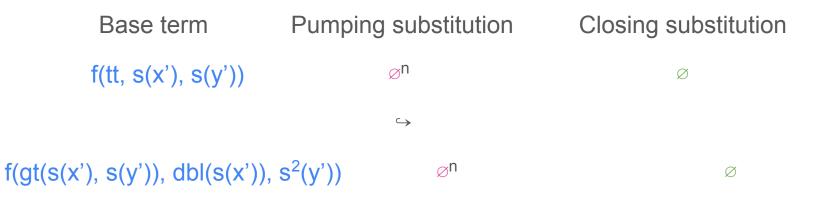
 $s \otimes^n \otimes \hookrightarrow t \otimes^n \otimes if s \to t \in R$



Creating Pattern Rules (11/16) Instantiation

 $f(tt, \textbf{x}, \textbf{y}) \rightarrow f(gt(\textbf{x}, \textbf{y}), dbl(\textbf{x}), s(\textbf{y}))$

We instantiate x to be s(x') and y to be s(y').



Creating Pattern Rules (12/16)

Instantiation of pumping substitutions

 $s \sigma^n \mu \hookrightarrow t \theta^n v$ if ρ commutes with

 $s (\sigma \rho)^n \mu \, \hookrightarrow \, t \, (\theta \rho)^n \, v \qquad \qquad \sigma, \, \mu, \, \theta, \, v$

 $f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y))$

Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ \varnothing \hookrightarrow $(gt(s(x'), s(y')), dbl(s(x')), s^2(y'))$ $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ \varnothing

Creating Pattern Rules (13/16)

Instantiation of closing substitutions

 $\mathbf{s} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu} \, \hookrightarrow \, \mathbf{t} \, \boldsymbol{\theta}^{\mathsf{n}} \, \mathbf{v}$

s σⁿ (μρ) → t θⁿ (νρ)

Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow f(gt(s(x'), s(y')), dbl(s(x')), s^2(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x), y' \mapsto 0]$

$$f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y))$$

39

Creating Pattern Rules (14/16) Narrowing

 $f(tt, \mathbf{x}, \mathbf{y}) \rightarrow f(gt(\mathbf{x}, \mathbf{y}), dbl(\mathbf{x}), s(\mathbf{y}))$

Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow

Creating Pattern Rules (15/16) Narrowing

 $\begin{array}{l} f(\text{tt, } \textbf{x}, \textbf{y}) \rightarrow f(\text{gt}(\textbf{x}, \textbf{y}), \, dbl(\textbf{x}), \, s(\textbf{y})) \\ dbl(\textbf{x}) \quad \rightarrow times(s(s(0)), \, \textbf{x}) \end{array}$

 $\frac{\mathbf{s} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu} \, \hookrightarrow \, \mathbf{t} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu} \, \hookrightarrow \, \mathbf{v} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu}}{\mathbf{s} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu} \, \hookrightarrow \, \mathbf{t} [\mathbf{v}]_{\pi} \, \boldsymbol{\sigma}^{\mathsf{n}} \, \boldsymbol{\mu}} \qquad \text{if } \mathbf{t} \mid_{\pi} = \mathbf{u}$

Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x), y' \mapsto 0]$ \hookrightarrow

 $f(tt, dbl(s(x')), s^{2}(y')) \qquad [x' \mapsto s(x'), y' \mapsto s(y')]^{n} \quad [x' \mapsto s(x), y' \mapsto 0]$

Creating Pattern Rules (16/16) Rewriting

 $\begin{array}{ll} f(tt,\,\textbf{x},\,\textbf{y}) \rightarrow f(gt(\textbf{x},\,\textbf{y}),\,dbl(\textbf{x}),\,s(\textbf{y})) \\ dbl(\textbf{x}) & \rightarrow times(s(s(0)),\,\textbf{x}) \end{array}$

For rewriting, we can just rewrite the right-hand side of the pattern rule with the rewriting rules from the original TRS.

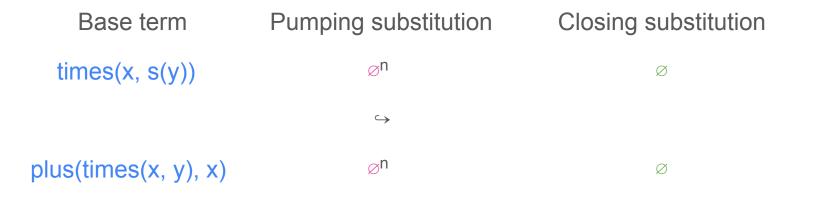
Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x), y' \mapsto 0]$

 \hookrightarrow

f(tt, times(s²(0), s(x')), s²(y')) [x' → s(x'), y' → s(y')]ⁿ [x' → s(x), y' → 0]

Creating Pattern Rules (16a/16c) times(x, s(y))→plus(times(x, y), x) Pattern rule from TRS

 $\mathbf{s} \otimes^{\mathsf{n}} \otimes \hookrightarrow \mathbf{t} \otimes^{\mathsf{n}} \otimes \quad \text{if } \mathbf{s} \to \mathbf{t} \in \mathbf{R}$



Creating Pattern Rules (16b/16c) $times(x, s(y)) \rightarrow plus(times(x, y), x)$

	$\mathbf{S} \otimes^{n} \otimes \hookrightarrow \mathbf{t} \otimes^{n} \otimes$	if s = t $ _{\pi}\sigma$ and
S	$\sigma^n \oslash \hookrightarrow \mathfrak{t}[\mathbf{z}]_{\pi} (\sigma \cup [\mathbf{z} \mapsto \mathfrak{t}[\mathbf{z}]_{\pi}])^n [\mathbf{z} \mapsto \mathfrak{t}[\mathbf{z}]_{\pi}]$	→ $t _{\pi}$] $z \in V$ is fresh

Base termPumping substitutionClosing substitutiontimes(x, s(y)) $[y \mapsto s(y)]^n$ \oslash \hookrightarrow \bigcirc plus(times(x, y), x) $[y \mapsto s(y), z \mapsto plus(z, x)]^n$ $[z \mapsto times(x, y)]$

Creating Pattern Rules (16c/16c) $f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y))$ Some steps further

After also narrowing with times and cleaning the pattern rule:

Base termPumping substitutionClosing substitutionf(tt, s(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(x'), y' \mapsto 0]$ \hookrightarrow

 $f(tt, s^{2}(x'), s^{2}(y')) \quad [x' \mapsto s^{2}(x'), y' \mapsto s(y')]^{n} \ [x' \mapsto times(s^{2}(0), s(x')), y' \mapsto 0]$

Detecting Non-Termination (1/4)

$$f(tt, \mathbf{x}, \mathbf{y}) \rightarrow f(gt(\mathbf{x}, \mathbf{y}), dbl(\mathbf{x}), s(\mathbf{y}))$$

Some steps further

After also narrowing with times and cleaning the pattern rule.

Check whether the pattern substitutions of the right-hand side are specializations of the pattern substitutions of the left-hand side

Base termPumping substitutionClosing substitutionf(tt, s²(x'), s(y'))
$$[x' \mapsto s(x'), y' \mapsto s(y')]^n$$
 $[y' \mapsto 0]$ \hookrightarrow f(tt, s³(x'), s²(y')) $[x' \mapsto s²(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(times(s²(0), x')), y' \mapsto 0]$

$\begin{array}{ll} \mbox{Detecting Non-Termination (2/4)} & f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y)) \\ \mbox{Theorem} \end{array}$

Let $\mathbf{s} \sigma^n \mu \hookrightarrow \mathbf{t} (\sigma^m \theta)^n (\mu \mathbf{v})$ be correct w.r.t. a TRS R, where θ commutes with both σ and μ . If there is a position π and some $b \in \mathbb{N}$ such that $\mathbf{s} \sigma^b = \mathbf{t}|_{\pi}$, then R is non-terminating.

 $\mathbf{s} \sigma^n \mu \rightarrow_R^+ \mathbf{t} (\sigma^m \theta)^n (\mu v)$ and if we zoom in on the subterm where $\mathbf{s} \sigma^b = \mathbf{t}|_{\pi}$ we get

 $t|_{\pi} (\sigma^{m} \theta)^{n} (\mu v) = s \sigma^{b} (\sigma^{m} \theta)^{n} (\mu v)$ $= s \sigma^{mn+b} \mu \theta^{n} v$

$\begin{array}{ll} \mbox{Detecting Non-Termination (3/4)} & f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y)) \\ \mbox{Theorem} \end{array}$

Let $\mathbf{s} \sigma^n \mu \hookrightarrow \mathbf{t} (\sigma^m \theta)^n \mu \mathbf{v}$ be correct w.r.t. a TRS R, where θ commutes with both σ and μ . If there is a position π and some $b \in \mathbb{N}$ such that $\mathbf{s} \sigma^b = \mathbf{t}|_{\pi}$, then R is non-terminating.

Base termPumping substitutionClosing substitutionf(tt, s²(x'), s(y')) $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[y' \mapsto 0]$ \hookrightarrow f(tt, s³(x'), s²(y')) $[x' \mapsto s²(x'), y' \mapsto s(y')]^n$ $[x' \mapsto s(times(s²(0), x')), y' \mapsto 0]$

$\begin{array}{ll} \mbox{Detecting Non-Termination (4/4)} & f(tt, x, y) \rightarrow f(gt(x, y), dbl(x), s(y)) \\ \mbox{Theorem} \end{array}$

Let $\mathbf{s} \sigma^n \mu \hookrightarrow \mathbf{t} (\sigma^m \theta)^n \mu v$ be correct w.r.t. a TRS R, where θ commutes with both σ and μ . If there is a position π and some $b \in \mathbb{N}$ such that $\mathbf{s} \sigma^b = \mathbf{t}|_{\pi}$, then R is non-terminating. And since $\mathbf{s} \sigma = \mathbf{f}(\mathbf{tt}, \mathbf{s}^3(\mathbf{x}'), \mathbf{s}^2(\mathbf{y}')) = \mathbf{f}(\mathbf{tt}, \mathbf{s}^3(\mathbf{x}'), \mathbf{s}^2(\mathbf{y}'))$, there is an infinite reduction sequence and the TRS is non-terminating.

Base termPumping substitutionClosing substitution $f(tt, s^2(x'), s(y'))$ $[x' \mapsto s(x'), y' \mapsto s(y')]^n$ $[y' \mapsto 0]$

 \hookrightarrow

 $f(tt, s^{3}(x'), s^{2}(y')) ([x' \mapsto s(x'), y' \mapsto s(y')][x' \mapsto s(x')])^{n} ([y' \mapsto 0][x' \mapsto s(times(s^{2}(0), x'))])$

Proving Non-Looping Non-Termination Automatically

Recap of the technique

- 1. Generate pattern rules from the TRS.
- 2. Modify to avoid empty pattern substitutions.
- 3. Prepare for narrowing $p \rightarrow q$ with $p' \rightarrow q'$:
 - a. Make the base term of q at some position equal to the base term of p'.
 - b. Make the pumping substitutions and closing substitutions of the two pattern rules equal.
- 4. Narrow $p \hookrightarrow q$ with $p' \hookrightarrow q'$.
- 5. Check for non-termination of the resulting rule $s \rightarrow t$: is t a specialization of s?

How is this automated?

Soundness of the inference rules

For all the inference rules of the form

 $\begin{array}{ccc} p_1 \hookrightarrow q_1 & \dots & p_k \hookrightarrow q_k \\ & & & \\ & & & \\ p \hookrightarrow q \end{array}$

If all pattern rules $p_1 \hookrightarrow q_1, ..., p_k \hookrightarrow q_k$ are correct w.r.t. a TRS R, then the pattern rule $p \hookrightarrow q$ is also correct w.r.t. R.

Final remarks

Proving Non-Looping Non-Termination Automatically

- The technique can prove non-termination of looping and non-looping TRS
- Tested on 58 typical non-looping non-terminating TRSs, and were able to prove non-termination in 75.9% of the non-looping examples
- There are TRSs for which the non-termination cannot be proved with the technique.