EuroProofNet

Properties of $\lambda\Pi/\mathcal{R}$

Frédéric Blanqui

Deduc⊢eam

Some important properties

TC	decidability of the typing relation
SN	termination of $\rightarrow_{\beta\mathcal{R}}$ from typable terms
SR_β	preservation of typing by $\rightarrow_\beta$
$\mathsf{SR}_\mathcal{R}$	preservation of typing by $\rightarrow_\mathcal{R}$
LCR	local confluence of $\rightarrow_{\beta\mathcal{R}}$ from arbitrary terms
CR	confluence of $\rightarrow_{\beta\mathcal{R}}$ from arbitrary terms
TCR	confluence of $\rightarrow_{\beta\mathcal{R}}$ from typable terms

Remarks:

- CR + SR $\Rightarrow$ TCR
- LCR + SN $\Rightarrow$ CR (Newman's Lemma)
- LCR + SN + SR $\Rightarrow$ TCR

Outline

Decidability of type-checking (TC)

Subject-reduction for β (SR$_\beta$)

Subject-reduction for rules (SR$_\mathcal{R}$)

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (SN)

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$(\text{conv})\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^*_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$\text{(conv)}\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^{*}_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

$$\text{(fun)}\ \frac{\Gamma \text{ valid}}{\Gamma \vdash f \Uparrow A_f} \qquad \text{(var)}\ \frac{\Gamma, x{:}A, \Gamma' \text{ valid}}{\Gamma, x{:}A, \Gamma' \vdash x \Uparrow A}$$

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$(\text{conv})\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^*_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

$$(\text{fun})\ \frac{\Gamma\ \text{valid}}{\Gamma \vdash f \Uparrow A_f} \qquad (\text{var})\ \frac{\Gamma, x{:}A, \Gamma'\ \text{valid}}{\Gamma, x{:}A, \Gamma' \vdash x \Uparrow A}$$

$$(\text{sort})\ \frac{\Gamma\ \text{valid}}{\Gamma \vdash \mathtt{TYPE} \Uparrow \mathtt{KIND}} \qquad (\text{prod})\ \frac{\Gamma \vdash A \Downarrow \mathtt{TYPE} \quad \Gamma, x{:}A \vdash B \Uparrow s}{\Gamma \vdash \Pi x{:}A.B \Uparrow s}$$

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$\text{(conv)}\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^*_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

$$\text{(fun)}\ \frac{\Gamma\ \text{valid}}{\Gamma \vdash f \Uparrow A_f} \qquad \text{(var)}\ \frac{\Gamma, x{:}A, \Gamma'\ \text{valid}}{\Gamma, x{:}A, \Gamma' \vdash x \Uparrow A}$$

$$\text{(sort)}\ \frac{\Gamma\ \text{valid}}{\Gamma \vdash \mathtt{TYPE} \Uparrow \mathtt{KIND}} \qquad \text{(prod)}\ \frac{\Gamma \vdash A \Downarrow \mathtt{TYPE} \quad \Gamma, x{:}A \vdash B \Uparrow s}{\Gamma \vdash \Pi x{:}A.B \Uparrow s}$$

$$\text{(abs)}\ \frac{\Gamma \vdash A \Downarrow \mathtt{TYPE} \quad \Gamma, x{:}A \vdash t \Uparrow B \quad B \neq \mathtt{KIND}}{\Gamma \vdash \lambda x{:}A.t \Uparrow \Pi x{:}A.B}$$

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$\text{(conv)}\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^{*}_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

$$\text{(fun)}\ \frac{\Gamma \text{ valid}}{\Gamma \vdash f \Uparrow A_f} \qquad \text{(var)}\ \frac{\Gamma, x{:}A, \Gamma' \text{ valid}}{\Gamma, x{:}A, \Gamma' \vdash x \Uparrow A}$$

$$\text{(sort)}\ \frac{\Gamma \text{ valid}}{\Gamma \vdash \mathtt{TYPE} \Uparrow \mathtt{KIND}} \qquad \text{(prod)}\ \frac{\Gamma \vdash A \Downarrow \mathtt{TYPE} \quad \Gamma, x{:}A \vdash B \Uparrow s}{\Gamma \vdash \Pi x{:}A.B \Uparrow s}$$

$$\text{(abs)}\ \frac{\Gamma \vdash A \Downarrow \mathtt{TYPE} \quad \Gamma, x{:}A \vdash t \Uparrow B \quad B \neq \mathtt{KIND}}{\Gamma \vdash \lambda x{:}A.t \Uparrow \Pi x{:}A.B}$$

$$\text{(app)}\ \frac{\Gamma \vdash t \Uparrow C \quad C \hookrightarrow^{*}_{\beta\mathcal{R}} \Pi x{:}A.B \quad \Gamma \vdash u \Downarrow A}{\Gamma \vdash tu \Uparrow B\{x \mapsto u\}}$$

Decidability of type-checking (TC)

mix type-inference $\Uparrow$ and type-checking $\Downarrow$

$$\text{(conv)}\ \frac{\Gamma \vdash t \Uparrow A \quad A \downarrow^*_{\beta\mathcal{R}} B}{\Gamma \vdash t \Downarrow B}$$

$$\boxed{\mathsf{SN} + \mathsf{LCR} + \mathsf{SR} \Rightarrow \mathsf{TC}}$$

$$\text{(app)}\ \frac{\Gamma \vdash t \Uparrow C \quad C \hookrightarrow^*_{\beta\mathcal{R}} \Pi x{:}A.B \quad \Gamma \vdash u \Downarrow A}{\Gamma \vdash tu \Uparrow B\{x \mapsto u\}}$$

Outline

Decidability of type-checking (TC)

Subject-reduction for β (SR_β)

Subject-reduction for rules ($\mathsf{SR}_\mathcal{R}$)

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (SN)

Type safety, aka subject-reduction (SR) in typed programming languages

assume a typed prog. language with operational semantics $\hookrightarrow$

subject-reduction property (SR):

$$\boxed{\text{if } t : A \text{ and } t \hookrightarrow u, \text{ then } u : A}$$

meaning: an expression checked of type A at compile time
can only evaluate to a value of type A

- fondamental property of *statically-typed* prog. languages
- ensure memory safety

SR in type-based logical systems

assume a type system with cut-elimination relation $\hookrightarrow$

subject-reduction property (SR):

$$\boxed{\text{if } t : A \text{ and } t \hookrightarrow u \text{, then } u : A}$$

meaning: a proof of proposition A can only reduce to a proof of A

- correctness of cut-elimination
- correctness of type inference in dependent type theories

Subject-reduction for β (SR$_\beta$)

$$\vdash (\lambda x\!:\!A, t)u : C$$

$$\Downarrow$$

$$\vdash t\{x \mapsto u\} : C \ ?$$

Subject-reduction for β (SR$_\beta$)

$$\dfrac{\dfrac{\vdash (\lambda x\!:\!A, t) : \Pi x\!:\!A', B' \qquad \vdash u : A'}{\vdash (\lambda x\!:\!A, t)u : B'\{x \mapsto u\}} \qquad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{\vdash (\lambda x\!:\!A, t)u : C}$$

$$\Downarrow$$

$$\vdash t\{x \mapsto u\} : C \ ?$$

Subject-reduction for β (SR$_\beta$)

$$\dfrac{\dfrac{\dfrac{\dfrac{x:A \vdash t : B}{\vdash (\lambda x:A, t) : \Pi x:A, B} \qquad \Pi x:A, B \downarrow^*_{\beta\mathcal{R}} \Pi x:A', B'}{\vdash (\lambda x:A, t) : \Pi x:A', B'} \qquad \vdash u : A'}{\vdash (\lambda x:A, t)u : B'\{x \mapsto u\}} \qquad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{\vdash (\lambda x:A, t)u : C}$$

$$\Downarrow$$

$$\vdash t\{x \mapsto u\} : C \ ?$$

Subject-reduction for β (SR_β)

$$\dfrac{\dfrac{\dfrac{\dfrac{x:A \vdash t:B}{\vdash (\lambda x:A,t) : \Pi x:A,B} \qquad \Pi x:A,B \downarrow^*_{\beta\mathcal{R}} \Pi x:A',B'}{\vdash (\lambda x:A,t) : \Pi x:A',B'} \qquad \vdash u:A'}{\vdash (\lambda x:A,t)u : B'\{x \mapsto u\}} \qquad B'\{x\mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{\vdash (\lambda x:A,t)u : C}$$

$$\Downarrow$$

$$\dfrac{\dfrac{x:A \vdash t:B \qquad u:A\ ?}{\vdash t\{x\mapsto u\} : B\{x\mapsto u\}} \qquad B\{x\mapsto u\} \downarrow^*_{\beta\mathcal{R}} C\ ?}{\vdash t\{x\mapsto u\} : C}$$

Subject-reduction for β (SR$_\beta$)

$$\dfrac{\dfrac{\dfrac{\dfrac{x:A\vdash t:B}{\vdash(\lambda x:A,t):\Pi x:A,B}\qquad \Pi x:A,B\downarrow^*_{\beta\mathcal{R}}\Pi x:A',B'}{\vdash(\lambda x:A,t):\Pi x:A',B'}\qquad \vdash u:A'}{\vdash(\lambda x:A,t)u:B'\{x\mapsto u\}}\qquad B'\{x\mapsto u\}\downarrow^*_{\beta\mathcal{R}} C}{\vdash(\lambda x:A,t)u:C}$$

$$\Downarrow$$

$$\dfrac{\dfrac{x:A\vdash t:B \qquad \dfrac{u:A' \qquad A'\downarrow^*_{\beta\mathcal{R}} A\ ?}{u:A}}{\vdash t\{x\mapsto u\}:B\{x\mapsto u\}} \qquad \dfrac{\dfrac{B\downarrow^*_{\beta\mathcal{R}} B'\ ?}{B\{x\mapsto u\}\downarrow^*_{\beta\mathcal{R}} B'\{x\mapsto u\}}\qquad B'\{x\mapsto u\}\downarrow^*_{\beta\mathcal{R}} C}{B\{x\mapsto u\}\downarrow^*_{\beta\mathcal{R}} C}}{\vdash t\{x\mapsto u\}:C}$$

Subject-reduction for β (SR$_\beta$)

$$\dfrac{\dfrac{\dfrac{\dfrac{x\!:\!A \vdash t : B}{\vdash (\lambda x\!:\!A, t) : \Pi x\!:\!A, B} \qquad \Pi x\!:\!A, B \downarrow^*_{\beta\mathcal{R}} \Pi x\!:\!A', B'}{\vdash (\lambda x\!:\!A, t) : \Pi x\!:\!A', B'} \qquad \vdash u : A'}{\vdash (\lambda x\!:\!A, t)u : B'\{x \mapsto u\}} \qquad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{\vdash (\lambda x\!:\!A, t)u : C}$$

$$\Downarrow$$

$$\dfrac{\dfrac{x\!:\!A \vdash t : B \qquad \dfrac{u : A' \qquad \dfrac{\Pi x\!:\!A, B \downarrow^*_{\beta\mathcal{R}} \Pi x\!:\!A', B'}{A' \downarrow^*_{\beta\mathcal{R}} A}}{u : A}}{\vdash t\{x \mapsto u\} : B\{x \mapsto u\}} \qquad \dfrac{\dfrac{\dfrac{\Pi x\!:\!A, B \downarrow^*_{\beta\mathcal{R}} \Pi x\!:\!A', B'}{B \downarrow^*_{\beta\mathcal{R}} B'}}{B\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} B'\{x \mapsto u\}} \qquad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{B\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}}{\vdash t\{x \mapsto u\} : C}$$

Subject-reduction for β (SR_β)

$$\dfrac{\dfrac{\dfrac{\dfrac{x:A \vdash t : B}{\vdash (\lambda x:A, t) : \Pi x:A, B} \quad \Pi x:A, B \downarrow^*_{\beta\mathcal{R}} \Pi x:A', B'}{\vdash (\lambda x:A, t) : \Pi x:A', B'} \quad \vdash u : A'}{\vdash (\lambda x:A, t)u : B'\{x \mapsto u\}} \quad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{\vdash (\lambda x:A, t)u : C}$$

$$\Downarrow$$

$$\dfrac{\dfrac{x:A \vdash t : B \quad \dfrac{u : A' \quad \dfrac{\Pi x:A, B \downarrow^*_{\beta\mathcal{R}} \Pi x:A', B'}{A' \downarrow^*_{\beta\mathcal{R}} A}}{u : A}}{\vdash t\{x \mapsto u\} : B\{x \mapsto u\}} \quad \dfrac{\dfrac{\dfrac{\Pi x:A, B \downarrow^*_{\beta\mathcal{R}} \Pi x:A', B'}{B \downarrow^*_{\beta\mathcal{R}} B'}}{B\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} B'\{x \mapsto u\}} \quad B'\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}{B\{x \mapsto u\} \downarrow^*_{\beta\mathcal{R}} C}}{\vdash t\{x \mapsto u\} : C}$$

$$\boxed{\mathrm{CR} \Rightarrow \mathrm{SR}_\beta}$$

Outline

Decidability of type-checking (TC)

Subject-reduction for β (SR_β)

Subject-reduction for rules ($\mathsf{SR}_\mathcal{R}$)

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (SN)

Subject-reduction (SR) for a rule $l \hookrightarrow r$

Goal: $\forall \Gamma, \sigma, C, \quad \Gamma \vdash l\sigma : C \quad \Rightarrow \quad \Gamma \vdash r\sigma : C \quad ?$

undecidable in $\lambda\Pi/\mathcal{R}$ [Saillard, 2015]

A first (not so good) idea

Goal: $\forall \Gamma, \sigma, C, \quad \Gamma \vdash l\sigma : C \quad \Rightarrow \quad \Gamma \vdash r\sigma : C \quad ?$

there exists B such that $l : B$ and $r : B$?

A first (not so good) idea

Goal: $\forall \Gamma, \sigma, C, \quad \Gamma \vdash l\sigma : C \quad \Rightarrow \quad \Gamma \vdash r\sigma : C \quad ?$

there exists B such that $l : B$ and $r : B$?

$\Rightarrow$ enforces many rules to be non-linear

$\Rightarrow$ rewriting is less efficient and confluence more difficult to prove

Example: tail function on vectors

```
symbol A:TYPE

symbol V:N → TYPE
symbol nil:V 0
symbol cons:A → Π n:N,V n → V(s n)

symbol tail:Π n:N,V(s n) → V n

rule tail $n (cons $x $p $v) ↪ $v
```

Example: tail function on vectors

```
symbol A:TYPE

symbol V:N → TYPE
symbol nil:V 0
symbol cons:A → Π n:N,V n → V(s n)

symbol tail:Π n:N,V(s n) → V n

rule tail $n (cons $x $p $v) ↪ $v
```

the LHS is not typable:

	`cons x p v`	has type	`V(s p)`
but	`tail n`	expects an argument of type	`V(s n)`

replacing `p` by `n` makes it typable but non-linear

Non-linearity breaks confluence on untyped terms

Assume that we have a rule $Dxx \hookrightarrow_{\mathcal{R}} E$ with E a constant

Then, $\hookrightarrow_{\beta} \cup \hookrightarrow_{\mathcal{R}}$ is not confluent on untyped terms

Non-linearity breaks confluence on untyped terms

Assume that we have a rule $Dxx \hookrightarrow_{\mathcal{R}} E$ with E a constant

Then, $\hookrightarrow_\beta \cup \hookrightarrow_{\mathcal{R}}$ is not confluent on untyped terms

$$\text{Take: } \begin{cases} F = \lambda c, \lambda a, Da(ca) \\ C = Y_F = (\lambda x, F(xx))(\lambda x, F(xx)) \hookrightarrow_\beta FC \\ A = Y_C = (\lambda x, C(xx))(\lambda x, C(xx)) \hookrightarrow_\beta CA \end{cases}$$

Then $A \hookrightarrow_\beta CA \hookrightarrow_\beta FCA \hookrightarrow_\beta^2 DA(CA) \hookrightarrow_\beta D(CA)(CA) \hookrightarrow_{\mathcal{R}} E$

Non-linearity breaks confluence on untyped terms

Assume that we have a rule $Dxx \hookrightarrow_{\mathcal{R}} E$ with E a constant

Then, $\hookrightarrow_\beta \cup \hookrightarrow_{\mathcal{R}}$ is not confluent on untyped terms

$$\text{Take: } \begin{cases} F = \lambda c, \lambda a, Da(ca) \\ C = Y_F = (\lambda x, F(xx))(\lambda x, F(xx)) \hookrightarrow_\beta FC \\ A = Y_C = (\lambda x, C(xx))(\lambda x, C(xx)) \hookrightarrow_\beta CA \end{cases}$$

Then $A \hookrightarrow_\beta CA \hookrightarrow_\beta FCA \hookrightarrow_\beta^2 DA(CA) \hookrightarrow_\beta D(CA)(CA) \hookrightarrow_{\mathcal{R}} E$

and thus $A \hookrightarrow_\beta CA \hookrightarrow_\beta^* CE$ too but

CE can never reduce to E ($CE \hookrightarrow_\beta FCE \hookrightarrow_\beta^2 DE(CE) \hookrightarrow_\beta \ldots$)

Example: tail function on vectors

```
symbol V:N → TYPE
symbol nil:V0
symbol cons:A → Π n:N,V n → V(s n)

symbol tail:Π n:N,V(s n) → V n
```

yet the rule preserves typing:

- let `tail n (cons x p v)` be a typable instance of the LHS
- by inversion of typing rules, we get:

$$\underbrace{\texttt{tail}\ \underbrace{\texttt{n}}_{:\texttt{N}}\ \underbrace{(\texttt{cons}\ \underbrace{\texttt{x}}_{:\texttt{A}}\ \underbrace{\texttt{p}}_{:\texttt{N}}\ \underbrace{\texttt{v}}_{:\texttt{V}\,\texttt{p}})}_{:\texttt{V}(\texttt{s}\,\texttt{p})\downarrow^*_{\beta\mathcal{R}}\texttt{V}(\texttt{s}\,\texttt{n})}}_{:\texttt{V}\,\texttt{n}} \hookrightarrow \underbrace{\texttt{v}}_{:\texttt{V}\,\texttt{p}}$$

- since `V` and `s` are undefined, $\texttt{V}(\texttt{s}\,\texttt{p}) \downarrow^*_{\beta\mathcal{R}} \texttt{V}(\texttt{s}\,\texttt{n})$ implies $\texttt{p} \downarrow^*_{\beta\mathcal{R}} \texttt{n}$

Procedure for checking SR

Step 1: compute the equations $\mathcal{E}$ that must be satisfied for the LHS to be of type C (fresh constant)

goal: prove that the RHS has type C modulo $\mathcal{E}$

problem: how to type-check modulo equations?

Procedure for checking SR

Step 1: compute the equations $\mathcal{E}$ that must be satisfied for the LHS to be of type C (fresh constant)

goal: prove that the RHS has type C modulo $\mathcal{E}$

problem: how to type-check modulo equations?

Step 2: turn the equations into a convergent rewrite system $\mathcal{S}$ using **Knuth-Bendix completion**

Step 3: check that the RHS has type C in $\lambda\Pi/\mathcal{R}+\mathcal{S}$

Knuth-Bendix completion (1969)

Knuth-Bendix completion consists in turning a set of equations $\mathcal{E}$ into a terminating and eventually confluent set of rewrite rules $\mathcal{R}$ having the same equational theory by:

- turning an equation $l = r$ into a rewrite rule $l \hookrightarrow r$ if $l > r$ in some fixed reduction ordering $>$
- turning a non-confluent critical pair between two overlapping rule left hand-hides into a new equation

this may not terminate!

Example of Knuth-Bendix completion

Take the equations:

1. $x + 0 = x$ 2. $x + (s\,y) = s(x + y)$ 3. $(x + y) + z = x + (y + z)$

Example of Knuth-Bendix completion

Take the equations:

1. $x + 0 = x$ 2. $x + (s\,y) = s(x + y)$ 3. $(x + y) + z = x + (y + z)$

The lexicographic path ordering $>$ with $+ > s > 0$ and comparison of arguments from right to left can orient all the equations from left to right:

1. $x + 0 \hookrightarrow x$ 2. $x + (s\,y) \hookrightarrow s(x + y)$ 3. $(x + y) + z \hookrightarrow x + (y + z)$

Example of Knuth-Bendix completion

Take the equations:

1. $x + 0 = x$ 2. $x + (s\, y) = s(x + y)$ 3. $(x + y) + z = x + (y + z)$

The lexicographic path ordering $>$ with $+ > s > 0$ and comparison of arguments from right to left can orient all the equations from left to right:

1. $x + 0 \hookrightarrow x$ 2. $x + (s\, y) \hookrightarrow s(x + y)$ 3. $(x + y) + z \hookrightarrow x + (y + z)$

But there are critical pairs. How many?

Example of Knuth-Bendix completion

Take the equations:

1. $x + 0 = x$ 2. $x + (s\, y) = s(x + y)$ 3. $(x + y) + z = x + (y + z)$

The lexicographic path ordering $>$ with $+ > s > 0$ and comparison of arguments from right to left can orient all the equations from left to right:

1. $x + 0 \hookrightarrow x$ 2. $x + (s\, y) \hookrightarrow s(x + y)$ 3. $(x + y) + z \hookrightarrow x + (y + z)$

But there are critical pairs. How many? 5

1. $x + z \;\; {}_1\!\hookleftarrow \;\; (x + 0) + z \;\; \hookrightarrow_3 \;\; x + (0 + z)$
2. $s(x + y) + z \;\; {}_2\!\hookleftarrow \;\; (x + s\, y) + z \;\; \hookrightarrow_3 \;\; x + (s\, y + z)$
3. $(x + (y + z)) + t \;\; {}_3\!\hookleftarrow \;\; ((x + y) + z) + t \;\; \hookrightarrow_3 \;\; (x + y) + (z + t)$
4. $x + y \;\; {}_1\!\hookleftarrow \;\; (x + y) + 0 \;\; \hookrightarrow_3 \;\; x + (y + 0)$
5. $s((x + y) + z) \;\; {}_2\!\hookleftarrow \;\; (x + y) + s\, z \;\; \hookrightarrow_3 \;\; x + (y + s\, z)$

Are they confluent?

Example of Knuth-Bendix completion

Take the equations:

1. $x + 0 = x$ 2. $x + (s\,y) = s(x + y)$ 3. $(x + y) + z = x + (y + z)$

The lexicographic path ordering $>$ with $+ > s > 0$ and comparison of arguments from right to left can orient all the equations from left to right:

1. $x + 0 \hookrightarrow x$ 2. $x + (s\,y) \hookrightarrow s(x + y)$ 3. $(x + y) + z \hookrightarrow x + (y + z)$

But there are critical pairs. How many? 5

1. $x + z \;\; {}_1\hookleftarrow \;\; (x + 0) + z \;\; \hookrightarrow_3 \;\; x + (0 + z)$
2. $s(x + y) + z \;\; {}_2\hookleftarrow \;\; (x + s\,y) + z \;\; \hookrightarrow_3 \;\; x + (s\,y + z)$
3. $(x + (y + z)) + t \;\; {}_3\hookleftarrow \;\; ((x + y) + z) + t \;\; \hookrightarrow_3 \;\; (x + y) + (z + t)$
4. $x + y \;\; {}_1\hookleftarrow \;\; (x + y) + 0 \;\; \hookrightarrow_3 \;\; x + (y + 0)$
5. $s((x + y) + z) \;\; {}_2\hookleftarrow \;\; (x + y) + s\,z \;\; \hookrightarrow_3 \;\; x + (y + s\,z)$

Are they confluent? Not 1, 2 and 3. This creates new equations:

4. $x + z = x + (0 + z)$ 5. $s(x + y) + z = x + (s\,y + z)$ ...

Step 1: compute typability constraints $\mathcal{E}$ of the LHS

$$\overbrace{\underbrace{t}_{\text{term}}}^{\text{input}} \uparrow \overbrace{\underbrace{A}_{\text{type}} \quad \underbrace{[\mathcal{E}]}_{\text{equations}}}^{\text{output}}$$

(var) $\dfrac{}{y \uparrow \widehat{y}\,[\emptyset]}$ ($\widehat{y}$ new constant for the unknown type of y)

(fun) $\dfrac{f : \Pi x_1{:}T_1, \ldots, \Pi x_n{:}T_n, U \quad t_1 \uparrow A_1[\mathcal{E}_1] \quad t_n \uparrow A_n[\mathcal{E}_n]}{f t_1 \ldots t_n \uparrow U\sigma[\mathcal{E}_1 \cup \ldots \cup \mathcal{E}_n \cup \{A_1 = T_1\sigma, \ldots, A_n = T_n\sigma\}]}$

where $\sigma = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$

$$\underbrace{\mathtt{tail}\ \underbrace{\mathtt{n}}_{\uparrow\widehat{\mathtt{n}}=\mathtt{N}}\ \underbrace{(\mathtt{cons}\ \underbrace{\mathtt{x}}_{\uparrow\widehat{\mathtt{x}}=\mathtt{A}}\ \underbrace{\mathtt{p}}_{\uparrow\widehat{\mathtt{p}}=\mathtt{N}}\ \underbrace{\mathtt{v}}_{\uparrow\widehat{\mathtt{v}}=\mathtt{V\,p}})}_{\uparrow\mathtt{V(s\,p)=V(s\,n)}}}_{\uparrow\mathtt{V\,n}}$$

Step 2: turn $\mathcal{E}$ into a convergent rewrite system $\mathcal{S}$

using Knuth-Bendix completion procedure (KB)
with any well-founded order total on ground terms (e.g. LPO)

remark: KB always terminates on ground equations in this case

<u>example:</u> $\widehat{\mathtt{x}} > \widehat{\mathtt{v}} > \widehat{\mathtt{p}} > \widehat{\mathtt{n}} > \mathtt{V} > \mathtt{T} > \mathtt{N} > \mathtt{s} > \mathtt{p} > \mathtt{n}$

$$\begin{array}{llllll} \mathcal{E}: & \widehat{\mathtt{x}} = \mathtt{A} & \widehat{\mathtt{p}} = \mathtt{N} & \widehat{\mathtt{v}} = \mathtt{V\,p} & \widehat{\mathtt{n}} = \mathtt{N} & \mathtt{V(s\,p)} = \mathtt{V(s\,n)} \\ \mathcal{S}: & \widehat{\mathtt{x}} \hookrightarrow \mathtt{A} & \widehat{\mathtt{p}} \hookrightarrow \mathtt{N} & \widehat{\mathtt{v}} \hookrightarrow \mathtt{V\,p} & \widehat{\mathtt{n}} \hookrightarrow \mathtt{N} & \mathtt{V(s\,p)} \hookrightarrow \mathtt{V(s\,n)} \end{array}$$

Step 3: check that RHS has same type as LHS modulo $\mathcal{S}$

$$\mathtt{tail}\ \underbrace{\underbrace{\mathtt{n}}_{\uparrow\widehat{\mathtt{n}}=\mathtt{N}}\ \underbrace{(\mathtt{cons}\ \underbrace{\mathtt{x}}_{\uparrow\widehat{\mathtt{x}}=\mathtt{A}}\ \underbrace{\mathtt{p}}_{\uparrow\widehat{\mathtt{p}}=\mathtt{N}}\ \underbrace{\mathtt{v}}_{\uparrow\widehat{\mathtt{v}}=\mathtt{V}\,\mathtt{p}})}_{\uparrow\mathtt{V}(\mathtt{s}\,\mathtt{p})=\mathtt{V}(\mathtt{s}\,\mathtt{n})}}_{\uparrow\mathtt{V}\,\mathtt{n}} \hookrightarrow \mathtt{v}$$

$$\mathcal{S}:\quad \widehat{\mathtt{x}} \hookrightarrow \mathtt{A} \quad \widehat{\mathtt{p}} \hookrightarrow \mathtt{N} \quad \widehat{\mathtt{v}} \hookrightarrow \mathtt{V}\,\mathtt{p} \quad \widehat{\mathtt{n}} \hookrightarrow \mathtt{N} \quad \mathtt{V}(\mathtt{s}\,\mathtt{p}) \hookrightarrow \mathtt{V}(\mathtt{s}\,\mathtt{n})$$

we now want to check if

$$\boxed{\mathtt{v} : \mathtt{V}\,\mathtt{n} \text{ modulo } \mathcal{S}\ ?}$$

Step 3: check that RHS has same type as LHS modulo $\mathcal{S}$

$$\underbrace{\texttt{tail}\ \underbrace{\texttt{n}}_{\uparrow\widehat{\texttt{n}}=\texttt{N}}\ \underbrace{(\texttt{cons}\ \underbrace{\texttt{x}}_{\uparrow\widehat{\texttt{x}}=\texttt{A}}\ \underbrace{\texttt{p}}_{\uparrow\widehat{\texttt{p}}=\texttt{N}}\ \underbrace{\texttt{v}}_{\uparrow\widehat{\texttt{v}}=\texttt{V}\,\texttt{p}})}_{\uparrow\texttt{V}(\texttt{s}\,\texttt{p})=\texttt{V}(\texttt{s}\,\texttt{n})}}_{\uparrow\texttt{V}\,\texttt{n}} \quad\hookrightarrow\quad \texttt{v}$$

$$\mathcal{S}: \quad \widehat{\texttt{x}} \hookrightarrow \texttt{A} \quad \widehat{\texttt{p}} \hookrightarrow \texttt{N} \quad \widehat{\texttt{v}} \hookrightarrow \texttt{V}\,\texttt{p} \quad \widehat{\texttt{n}} \hookrightarrow \texttt{N} \quad \texttt{V}(\texttt{s}\,\texttt{p}) \hookrightarrow \texttt{V}(\texttt{s}\,\texttt{n})$$

we now want to check if

$$\boxed{\texttt{v} : \texttt{V}\,\texttt{n} \text{ modulo } \mathcal{S}\ ?}$$

no it doesn't work since $\texttt{v} : \widehat{\texttt{v}}$ and $\widehat{\texttt{v}} \not\hookrightarrow^*_{\beta\mathcal{R}\mathcal{S}} \texttt{V}\,\texttt{n}$ 🙁

Step 1': simplify equations using confluence of $\hookrightarrow_{\beta\mathcal{R}}$

$$\mathcal{E}: \quad \widehat{\mathtt{x}} = \mathtt{A} \quad \widehat{\mathtt{p}} = \mathtt{N} \quad \widehat{\mathtt{v}} = \mathtt{Vp} \quad \widehat{\mathtt{n}} = \mathtt{N} \quad \mathtt{V(s\,p)} = \mathtt{V(s\,n)}$$

because V and s are undefined, hence injective, $\mathcal{E}$ is equivalent to:

$$\mathcal{E}': \quad \widehat{\mathtt{x}} = \mathtt{A} \quad \widehat{\mathtt{p}} = \mathtt{N} \quad \widehat{\mathtt{v}} = \mathtt{V\,p} \quad \widehat{\mathtt{n}} = \mathtt{N} \quad \mathtt{p} = \mathtt{n}$$

step 3 (KB) with $\widehat{\mathtt{x}} > \widehat{\mathtt{v}} > \widehat{\mathtt{p}} > \widehat{\mathtt{n}} > \mathtt{V} > \mathtt{T} > \mathtt{N} > \mathtt{s} > \mathtt{p} > \mathtt{n}$:

$$\mathcal{S}': \quad \widehat{\mathtt{x}} \hookrightarrow \mathtt{A} \quad \widehat{\mathtt{p}} \hookrightarrow \mathtt{N} \quad \widehat{\mathtt{v}} \hookrightarrow \mathtt{V\,n} \quad \widehat{\mathtt{n}} \hookrightarrow \mathtt{N} \quad \mathtt{p} \hookrightarrow \mathtt{n}$$

Step 3: check that RHS has same type as LHS modulo $\mathcal{S}$

$$\underbrace{\texttt{tail}\ \underbrace{\texttt{n}}_{\uparrow\widehat{\texttt{n}}=\texttt{N}}\ \underbrace{(\texttt{cons}\ \underbrace{\texttt{x}}_{\uparrow\widehat{\texttt{x}}=\texttt{A}}\ \underbrace{\texttt{p}}_{\uparrow\widehat{\texttt{p}}=\texttt{N}}\ \underbrace{\texttt{v}}_{\uparrow\widehat{\texttt{v}}=\texttt{V}\,\texttt{p}})}_{\uparrow\texttt{V}(\texttt{s}\,\texttt{p})=\texttt{V}(\texttt{s}\,\texttt{n})}}_{\uparrow\texttt{V}\,\texttt{n}} \quad\hookrightarrow\quad \texttt{v}$$

$$\mathcal{S}' : \quad \widehat{\texttt{x}} \hookrightarrow \texttt{A} \quad \widehat{\texttt{p}} \hookrightarrow \texttt{N} \quad \widehat{\texttt{v}} \hookrightarrow \texttt{Vn} \quad \widehat{\texttt{n}} \hookrightarrow \texttt{N} \quad \texttt{p} \hookrightarrow \texttt{n}$$

we want to check if

$$\boxed{\texttt{v} : \texttt{Vn} \text{ modulo } \mathcal{S}' \text{ ?}}$$

now it works since $\texttt{v} : \widehat{\texttt{v}}$ and $\widehat{\texttt{v}} \hookrightarrow \texttt{Vn}$ 🙂

Conclusion: procedure for $\mathrm{SR}(l \hookrightarrow r)$

A procedure to prove that a rewrite rule preserves typing in $\lambda\Pi/\mathcal{R}$:

Step 1: compute the equations $\mathcal{E}$ that must be satisfied for the LHS to be of type C (fresh constant)

Step 2: simplify equations using confluence of $\hookrightarrow_{\beta\mathcal{R}}$

Step 3: turn the equations into a convergent rewrite system $\mathcal{S}$ using Knuth-Bendix completion

Step 4: check that the RHS has type C in some sub-system of $\lambda\Pi/\mathcal{R} + \mathcal{S}$

$$\boxed{\mathrm{CR} + \mathrm{TC}^- \ \Rightarrow \mathrm{SR}_{\mathcal{R}}}$$

problem: confluence and termination of $\hookrightarrow_{\beta\mathcal{R}} \cup \hookrightarrow_{\mathcal{S}}$?

Outline

Decidability of type-checking (TC)

Subject-reduction for β (SR_β)

Subject-reduction for rules ($\mathsf{SR}_\mathcal{R}$)

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (SN)

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (1st attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then t is SN.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x : A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \quad \Rightarrow \quad \frac{t\ \text{SN} \quad u\ \text{SN}}{tu\ \text{SN?}}$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (1st attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then t is SN.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \quad \Rightarrow \quad \frac{t \text{ SN} \quad u \text{ SN}}{tu \text{ SN?}}$$

Can't we take $t = u = \lambda x\!:\!A, xx$?

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (1st attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then t is SN.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \quad \Rightarrow \quad \frac{t \text{ SN} \quad u \text{ SN}}{tu \text{ SN?}}$$

Can't we take $t = u = \lambda x{:}A, xx$? No, t is not typable.
But can't we find a similar example that is typable?

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (1st attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then t is SN.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \quad \Rightarrow \quad \frac{t \text{ SN} \quad u \text{ SN}}{tu \text{ SN?}}$$

Can't we take $t = u = \lambda x\!:\!A, xx$? No, t is not typable.
But can't we find a similar example that is typable?

$\Sigma = A : \texttt{TYPE}, c : (A \to A) \to A, f : A \to (A \to A)$
$\mathcal{R} = \{f(cx) \hookrightarrow x\}$
$t = \lambda x\!:\!A, fxx$
$u = ct$

Then $tu \hookrightarrow_\beta f(ct)(ct) \hookrightarrow_\mathcal{R} tu$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (1st attempt)

Conclusion: to prove the termination of an application, the termination of the function and of the argument is not enough

We need to prove a stronger property, **super-termination**: a term $t : \Pi x\!:\!A, B$ is super-terminating if, for all super-terminating argument $u : A$, $tu : B_x^u$ is super-terminating

As a consequence, we need to:

- interpret each type A by a set $[\![A]\!]$ of super-terminating terms
- prove that $t : A \Rightarrow t \in [\![A]\!]$

remark: super-termination is more usually called convertibility (Tait), reducibility (Girard) or computability (Stenlund)

Definition of super-termination (1st attempt)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T = \Pi x : A, B \\ \text{SN otherwise} \end{cases}$$

Is it well defined?

Definition of super-termination (1st attempt)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T = \Pi x\!:\!A, B \\ \text{SN otherwise} \end{cases}$$

Is it well defined?

Yes. By **Markowsky fixpoint theorem** (1976): every monotone function F on a chain-complete poset (every totally ordered subset has a lub) has a least fixpoint.

- The set $\mathcal{I} = \mathcal{F}_p(\mathcal{T}, \mathcal{P}(\mathcal{T}))$ of partial functions from $\mathcal{T}$ to its powerset is chain-complete wrt function extension $\subseteq$.
- The function $F : \mathcal{I} \to \mathcal{I}$ such that

$$F(I)(T) = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in I(A), tu \in I(B_x^u)\} \text{ if } T = \Pi x\!:\!A, B \\ \text{SN otherwise} \end{cases}$$

$$dom(F(I)) = \{T \mid T = \Pi x\!:\!A, B \Rightarrow A \in \mathrm{dom}(I) \wedge \forall u \in I(A), B_x^u \in \mathrm{dom}(I)\}$$

is monotone

Definition of super-termination (1st attempt)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T = \Pi x\!:\!A, B \\ \text{SN otherwise} \end{cases}$$

Does super-termination imply termination: $[\![T]\!] \subseteq SN$?

Definition of super-termination (1st attempt)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T = \Pi x : A, B \\ \text{SN otherwise} \end{cases}$$

Does super-termination imply termination: $[\![T]\!] \subseteq SN$?

Yes, if $[\![A]\!] \neq \emptyset$ whenever $T = \Pi x : A, B$.

Do we have $[\![T]\!] \neq \emptyset$?

Definition of super-termination (1st attempt)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \left\{ \begin{array}{l} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T = \Pi x\!:\!A, B \\ \text{SN otherwise} \end{array} \right.$$

Does super-termination imply termination: $[\![T]\!] \subseteq SN$?

Yes, if $[\![A]\!] \neq \emptyset$ whenever $T = \Pi x\!:\!A, B$.

Do we have $[\![T]\!] \neq \emptyset$?

Yes: for all T, $\{xu_1 \ldots u_n \mid x \in Var, u_1, \ldots, u_n \in SN\} \subseteq [\![T]\!]$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (2nd attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t \in [\![\Pi x\!:\!A, B]\!] \quad u \in [\![A]\!]}{tu \in [\![B_x^u]\!]?}$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (2nd attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t \in [\![\Pi x{:}A, B]\!] \quad u \in [\![A]\!]}{tu \in [\![B_x^u]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x{:}A, t : \Pi x{:}A, B} \Rightarrow \frac{t \in [\![B]\!]}{\lambda x{:}A, t \in [\![\Pi x{:}A, B]\!]?}$$

$$\forall u \in [\![A]\!], (\lambda x{:}A, t)u \in [\![B_x^u]\!]?$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (2nd attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t \in [\![\Pi x\!:\!A, B]\!] \quad u \in [\![A]\!]}{tu \in [\![B_x^u]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t \in [\![B]\!]}{\lambda x\!:\!A, t \in [\![\Pi x\!:\!A, B]\!]?}$$

$$\forall u \in [\![A]\!], (\lambda x\!:\!A, t)u \in [\![B_x^u]\!]?$$

A term is **neutral** if it is neither an abstraction nor a partially applied function symbol. Examples: $(\lambda x\!:\!A, t)u$ and $t + u$.

Lemma: a neutral term is super-terminating if all its reducts are super-terminating.

Proof. Since t is neutral, tu is not reducible at the top and $\hookrightarrow(tu) = \hookrightarrow(t)u \cup t \hookrightarrow(u)$.

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (2nd attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t \in [\![\Pi x\!:\!A, B]\!] \quad u \in [\![A]\!]}{tu \in [\![B_x^u]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t \in [\![B]\!]}{\lambda x\!:\!A, t \in [\![\Pi x\!:\!A, B]\!]?}$$

$$\forall u \in [\![A]\!], (\lambda x\!:\!A, t)u \in [\![B_x^u]\!]?$$

$$\forall u \in [\![A]\!], t_x^u \in [\![B_x^u]\!]?$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (2nd attempt)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t \in [\![\Pi x\!:\!A, B]\!] \quad u \in [\![A]\!]}{tu \in [\![B_x^u]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t \in [\![B]\!]}{\lambda x\!:\!A, t \in [\![\Pi x\!:\!A, B]\!]?}$$

$$\forall u \in [\![A]\!], (\lambda x\!:\!A, t)u \in [\![B_x^u]\!]?$$

$$\forall u \in [\![A]\!], t_x^u \in [\![B_x^u]\!]?$$

We need to generalize the theorem again:

A substitution σ is super-terminating wrt Γ, written $\sigma \models \Gamma$,
if, for all $(x, A) \in \Gamma$, $x\sigma \in [\![A\sigma]\!]$.

Theorem: for all Γ, t, A, σ, if $\Gamma \vdash t : A$ and $\sigma \models \Gamma$ then $t\sigma \in [\![A\sigma]\!]$.

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (3rd attempt)

Theorem: for all Γ, t, A, σ, if $\Gamma \vdash t : A$ and $\sigma \models \Gamma$ then $t\sigma \in [\![A\sigma]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

Yes since $B_x^u\sigma = B\sigma_x^{u\sigma}$.

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (3rd attempt)

Theorem: for all Γ, t, A, σ, if $\Gamma \vdash t : A$ and $\sigma \models \Gamma$ then $t \in [\![A\sigma]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x{:}A, t : \Pi x{:}A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x{:}A\sigma, t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!]?}$$

$$\forall u \in [\![A\sigma]\!], (\lambda x{:}A\sigma, t\sigma)u \in [\![B\sigma_x^u]\!]?$$

$$\forall u \in [\![A\sigma]\!], t\sigma_x^u \in [\![B\sigma_x^u]\!]?$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (3rd attempt)

Theorem: for all Γ, t, A, σ, if $\Gamma \vdash t : A$ and $\sigma \models \Gamma$ then $t \in [\![A\sigma]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x\!:\!A\sigma, t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!]?}$$

$$\text{(conv)}\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (3rd attempt)

Theorem: for all Γ, t, A, σ, if $\Gamma \vdash t : A$ and $\sigma \models \Gamma$ then $t \in [\![A\sigma]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x\!:\!A\sigma, t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!]?}$$

$$\text{(conv)}\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

No, we need $[\![\ ]\!]$ to be invariant by $\downarrow_{\beta\mathcal{R}}$.

Definition of super-termination (2nd attempt)

assuming that $\hookrightarrow_{\beta\mathcal{R}}$ is locally-confluent (LCR)

Let $\mathcal{T}$ be the set of terms.

$$[\![T]\!] = \begin{cases} \{t \in \mathcal{T} \mid \forall u \in [\![A]\!], tu \in [\![B_x^u]\!]\} \text{ if } T \in SN \wedge nf(T) = \Pi x\!:\!A, B \\ \text{SN otherwise} \end{cases}$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (4th attempt)

assuming that $\hookrightarrow_{\beta\mathcal{R}}$ is locally-confluent (LCR)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u \sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x{:}A, t : \Pi x{:}A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x{:}A\sigma, t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!]?}$$

$$\text{(conv)}\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (4th attempt)

assuming that $\hookrightarrow_{\beta\mathcal{R}}$ is locally-confluent (LCR)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x{:}A, t : \Pi x{:}A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x{:}A\sigma, t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!]?}$$

$$\text{(conv)}\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

Yes because $A\sigma \in SN$, $B\sigma \in SN$ and $nf(A\sigma) = nf(B\sigma)$.

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (4th attempt)

assuming that $\hookrightarrow_{\beta\mathcal{R}}$ is locally-confluent (LCR)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$\text{(app)}\ \frac{\Gamma \vdash t : \Pi x{:}A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$\text{(abs)}\ \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x{:}A, t : \Pi x{:}A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x{:}A\sigma, t\sigma \in [\![\Pi x{:}A\sigma, B\sigma]\!]?}$$

$$\text{(conv)}\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

$$\text{(sig)}\ \frac{f : A \in \Sigma \quad \vdash A : s}{\vdash f : A} \Rightarrow f \in [\![A]\!]?$$

Termination of $\hookrightarrow_{\beta\mathcal{R}}$ (4th attempt)

assuming that $\hookrightarrow_{\beta\mathcal{R}}$ is locally-confluent (LCR)

Theorem: for all Γ, t, A, if $\Gamma \vdash t : A$ then $t \in [\![A]\!]$.

Proof. By induction on the definition of $\vdash$.

$$(\text{app})\ \frac{\Gamma \vdash t : \Pi x\!:\!A, B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B_x^u} \Rightarrow \frac{t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!] \quad u\sigma \in [\![A\sigma]\!]}{(tu)\sigma \in [\![B_x^u\sigma]\!]?}$$

$$(\text{abs})\ \frac{\Gamma, x\!:\!A \vdash t : B}{\Gamma \vdash \lambda x\!:\!A, t : \Pi x\!:\!A, B} \Rightarrow \frac{t\sigma_x^u \in [\![B\sigma_x^u]\!]}{\lambda x\!:\!A\sigma, t\sigma \in [\![\Pi x\!:\!A\sigma, B\sigma]\!]?}$$

$$(\text{conv})\ \frac{\Gamma \vdash t : A \quad \Gamma \vdash A : s \quad A \downarrow_{\beta\mathcal{R}} B \quad \Gamma \vdash B : s}{\Gamma \vdash t : B} \Rightarrow \frac{t\sigma \in [\![A\sigma]\!]}{t\sigma \in [\![B\sigma]\!]?}$$

$$(\text{sig})\ \frac{f : A \in \Sigma \quad \vdash A : s}{\vdash f : A} \Rightarrow f \in [\![A]\!]?$$

to prove the super-termination of function symbols, we can use dependency pairs

Dependency pairs on first-order terms

dependency pairs: $fl_1 \dots l_i > gm_1 \dots m_j$ iff $fl_1 \dots l_i \hookrightarrow r \in \mathcal{R}$, $gm_1 \dots m_j$ is a subterm of r, $m_1 \dots m_j$ are all the arguments to which g is applied, and g is defined.

chain relation on terms $ft_1 \dots t_i$ with $t_1, \dots, t_i$ terminating:

$$\frac{t_1 \hookrightarrow^* l_1\sigma \quad \dots \quad t_i \hookrightarrow^* l_i\sigma \quad fl_1 \dots l_i > gm_1 \dots m_j}{ft_1 \dots t_i \mathrel{\tilde{>}} gm_1\sigma \dots m_j\sigma}$$

Theorem (Arts & Giesl 2000, reformulated):
function symbols are super-terminating if $\tilde{>}$ terminates

Dependency pairs in $\lambda\Pi/\mathcal{R}$

dependency pairs: idem

chain relation on terms $ft_1 \dots t_i$ with $t_1, \dots, t_i$ super-terminating:

$$\frac{t_1 \hookrightarrow^* l_1\sigma \quad \dots \quad t_i \hookrightarrow^* l_i\sigma \quad fl_1 \dots l_i > gm_1 \dots m_j}{ft_1 \dots t_i t_{i+1} \dots t_p \;\tilde{>}\; gm_1\sigma \dots m_j\sigma u_{j+1} \dots u_q}$$

Theorem: function symbols are super-terminating if $\tilde{>}$ terminates and the theory $(\Sigma, \mathcal{R})$ is well-structured and accessible

Well-structured theory

a theory $(\Sigma, \mathcal{R})$ is **well-structured** if:

- the strict part of the dependency relation $f \succeq g$ if g occurs in the type of f or in a right hand-side of a rule of f is well-founded (always true when Σ is finite)

Well-structured theory

a theory $(\Sigma, \mathcal{R})$ is **well-structured** if:

- the strict part of the dependency relation $f \succeq g$ if g occurs in the type of f or in a right hand-side of a rule of f is well-founded (always true when Σ is finite)

- for every rule $fl_1 \dots l_n \hookrightarrow r \in \mathcal{R}$ with $f : \Pi x_1 : A_1, \dots, \Pi x_n : A_n, B$, there is a typing environment Δ such that:

$$\boxed{\Delta \vdash_{fl_1 \dots l_n} r : B_{x_1}^{l_1} \dots_{x_n}^{l_n}}$$

  where $\vdash_{fl_1 \dots l_n}$ is similar to $\vdash$ except that types can only be typed using symbols $\prec f$

Accessible theory

a well-structured theory $(\Sigma, \mathcal{R})$ is **accessible** if, for every rule $fl_1 \ldots l_n \hookrightarrow r \in \mathcal{R}$, with $f : \Pi x_1 : A_1, \ldots, \Pi x_n : A_n, B$,

$$\boxed{\sigma \models \Delta \text{ whenever } {}^{l_1}_{x_1} \ldots {}^{l_n}_{x_n} \sigma \models x_1 : A_1, \ldots, x_n : A_n}$$

(matching preserves super-termination)

example of non-accessible pattern:

$$c\, y \quad \text{with} \quad c : (A \to B) \to A$$

$c(\lambda x, xx) \in [\![A]\!] = SN$ but $\lambda x, xx \notin [\![A \to B]\!]$

Termination of the chain relation $\tilde{>}$

there exist various techniques for proving the termination of a chain relation for first or simply-typed higher-order rewriting

a simple one is size-change termination (SCT)

Theorem: $\tilde{>}$ terminates if Σ is finite and, in the transitive closure of the graph on Σ having, for each dp $fl_1 \ldots l_p > gm_1 \ldots m_q$, an edge from f to g labeled by the matrix $(a_{ij})_{i \le p, j \le q}$ with

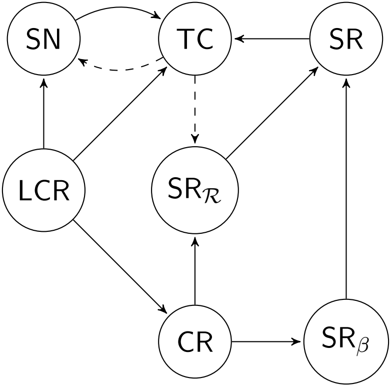
$$a_{ij} = \begin{cases} -1 & \text{if } l_i \rhd m_j \\ 0 & \text{if } l_i = m_j \\ +\infty & \text{otherwise} \end{cases}$$

all idempotent matrices labeling a loop has some -1 on the diagonal

Conclusion for termination

$$\boxed{\mathsf{LCR} + \mathsf{TC}^{-} \Rightarrow \mathsf{SN}}$$

Dependencies between properties



- - → for dependency on a sub-system