Confluence via strong normalisation in an algebraic λ-calculus with rewriting

– Accepted in LSFA 2011. To appear in EPTCS –

Pablo Buiras[1] Alejandro Díaz-Caro[2] Mauro Jaskelioff[1,3]

[1]Universidad Nacional de Rosario, FCEIA, Argentina

[2]Université de Grenoble, LIG, France

[3]CIFASIS-CONICET, Argentina

Recontre QuAND • July 18, 2011 • Marseille, France

$$M, N ::= x \mid \lambda x.M \mid (M)N \mid M + N \mid \alpha.M \mid 0$$

Beta reduction:

$$(\lambda x.M)N \to M[x := N]$$

"Algebraic" reductions:

$$\alpha.M + \beta.M \to (\alpha + \beta).M,$$
$$(M)(N_1 + N_2) \to (M)N_1 + (M)N_2,$$
$$\vdots$$

(oriented version of the axioms of vectorial spaces)

Two origins:

- Differential λ-calculus: capturing linearity *à la* Linear Logic

$\to$ *Removing the differential operator*: Algebraic λ-calculus (λ_{alg}) [Vaux'09]

- Quantum computing: superposition of programs

$\to$ *Linearity as in algebra*: Linear-algebraic λ-calculus (λ_{lin})

[Arrighi,Dowek'08]

$$M, N ::= x \mid \lambda x.M \mid (M)N \mid M + N \mid \alpha.M \mid 0$$

Beta reduction:

$$(\lambda x.M)N \to M[x := N]$$

“Algebraic” reductions:

$$\alpha.M + \beta.M \to (\alpha + \beta).M,$$

$$(M)(N_1 + N_2) \to (M)N_1 + (M)N_2,$$

...

...

...

(oriented version of the axioms of vectorial spaces)

	λ_{alg}	λ_{lin}
Origin	Linear Logic	Quantum computing
Strategy	Call-by-name	Call-by-value

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccc}
Y_B + (-1).Y_B & \longrightarrow (1-1).Y_B & \longrightarrow^* 0 \\
\downarrow & & \\
B + Y_B + (-1).Y_B & & \\
\downarrow_* & & \\
B & &
\end{array}$$

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccccc} Y_B + (-1).Y_B & \longrightarrow & (1-1).Y_B & \longrightarrow^* & 0 \\ \downarrow & & & & \\ B + Y_B + (-1).Y_B & & & & \\ \downarrow_* & & & & \\ B & & & & \end{array}$$

Solution 1 ($\lambda_{\mathbf{lin}}$):

$\alpha.M + \beta.M \to (\alpha + \beta).M$
only if M is closed-normal
and others similar restrictions

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccc} Y_B + (-1).Y_B & \longrightarrow (1-1).Y_B & \longrightarrow^* 0 \\ \downarrow & & \\ B + Y_B + (-1).Y_B & & \\ \downarrow_* & & \\ B & & \end{array}$$

Solution 1 ($\lambda_{\mathbf{lin}}$):
$\alpha.M + \beta.M \to (\alpha + \beta).M$
only if M is closed-normal
and others similar restrictions

Solution 2 ($\lambda_{\mathbf{alg}}$):

Only positive numbers

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccc} Y_B + (-1).Y_B & \longrightarrow (1-1).Y_B & \longrightarrow^* 0 \\ \downarrow & & \\ B + Y_B + (-1).Y_B & & \\ \downarrow_* & & \\ B & & \end{array}$$

Solution 1 ($\lambda_{\mathbf{lin}}$):

$\alpha.M + \beta.M \to (\alpha + \beta).M$
only if M is closed-normal
and others similar restrictions

Solution 2 ($\lambda_{\mathbf{alg}}$):

Only positive numbers

$$\begin{array}{ccc} Y_B + Y_B & \longrightarrow B + Y_B + Y_B & \longrightarrow B + 2.Y_B \\ \downarrow & & \\ 2.Y_B & & \end{array}$$

... and equalities

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccc} Y_B + (-1).Y_B & \longrightarrow (1-1).Y_B & \longrightarrow^* 0 \\ \downarrow & & \\ B + Y_B + (-1).Y_B & & \\ \downarrow_* & & \\ B & & \end{array}$$

Solution 1 ($\lambda_{\mathbf{lin}}$):
$\alpha.M + \beta.M \to (\alpha + \beta).M$
only if M is closed-normal
and others similar restrictions

Solution 2 ($\lambda_{\mathbf{alg}}$):

Only positive numbers

$$\begin{array}{ccc} Y_B + Y_B & \longrightarrow B + Y_B + Y_B & \longrightarrow B + 2.Y_B \\ \downarrow & & \\ 2.Y_B & & \end{array}$$

... and equalities

$\boxed{\text{Objective: to forbid } \infty - \infty}$

Confluence issues

$$Y_B = (\lambda x.(B + (x)x))\lambda x.(B + (x)x)$$

$$Y_B \to B + Y_B \to B + B + Y_B \to \dots$$

$$\begin{array}{ccc} Y_B + (-1).Y_B & \longrightarrow (1-1).Y_B & \longrightarrow^* 0 \\ \downarrow & & \\ B + Y_B + (-1).Y_B & & \\ \downarrow_* & & \\ B & & \end{array}$$

Solution 1 ($\lambda_{\mathbf{lin}}$):
$\alpha.M + \beta.M \to (\alpha + \beta).M$
only if M is closed-normal
and others similar restrictions

Solution 2 ($\lambda_{\mathbf{alg}}$):

Only positive numbers

$$\begin{array}{ccc} Y_B + Y_B & \longrightarrow B + Y_B + Y_B & \longrightarrow B + 2.Y_B \\ \downarrow & & \\ 2.Y_B & & \end{array}$$

... and equalities

Solution 3:
forbid ∞! (type system)

$\boxed{\text{Objective: to forbid } \infty - \infty}$

Possible type systems

Straightforward extension of a classic type system

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : T}{\Gamma \vdash M + N : T} +_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : T} \alpha_I \qquad \frac{}{\Gamma \vdash \mathbf{0} : T} 0_I$$

Pros:

- Simple

Cons:

- Too restrictive

Possible type systems

Straightforward extension of a classic type system

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : T}{\Gamma \vdash M + N : T} +_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : T} \alpha_I \qquad \frac{}{\Gamma \vdash \mathbf{0} : T} 0_I$$

Pros:

- Simple

Cons:

- Too restrictive

Scalar **type system** [Arrighi, Díaz-Caro 2009]

$$\frac{\Gamma \vdash M : \alpha.T \qquad \Gamma \vdash N : \beta.T}{\Gamma \vdash M + N : (\alpha + \beta).T} +_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T} \alpha_I \qquad \frac{}{\Gamma \vdash \mathbf{0} : 0.T} 0_I$$

Pros:

- Characterises the 'amount' of terms

Cons:

- Still too restrictive

Possible type systems

Additive **type system** [Díaz-Caro, Petit 2010]

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : R}{\Gamma \vdash M + N : T + R}\, +_I \qquad \cancel{\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T}\, \alpha_I} \qquad \frac{}{\Gamma \vdash \mathbf{0} : \overline{0}}\, 0_I$$

Pros:

- More versatile
- Interpretation in System F with pairs

Cons:

- Defined for a fragment (no scalars!)

Possible type systems

Additive **type system** `[Díaz-Caro, Petit 2010]`

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : R}{\Gamma \vdash M + N : T + R}\, +_I \qquad \cancel{\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T}\, \alpha_I} \qquad \frac{}{\Gamma \vdash \mathbf{0} : \overline{0}}\, 0_I$$

Pros:

- More versatile
- Interpretation in System F with pairs

Cons:

- Defined for a fragment (no scalars!)

Vectorial **type system** `[Arrighi, Díaz-Caro, Valiron 2011]`

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : R}{\Gamma \vdash M + N : T + R}\, +_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T}\, \alpha_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \mathbf{0} : 0.T}\, 0_I$$

Pros:

- Much more versatile
- Good characterisation of terms

Cons:

- Complex
- Overkill (confluence!)

Possible type systems

Additive **type system** `[Díaz-Caro, Petit 2010]`

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : R}{\Gamma \vdash M + N : T + R} +_I \qquad \cancel{\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T} \alpha_I} \qquad \frac{}{\Gamma \vdash \mathbf{0} : \overline{0}} 0_I$$

Pros:

- More versatile
- Interpretation in System F with pairs

Cons:

- Defined for a fragment (no scalars!)

Vectorial **type system** `[Arrighi, Díaz-Caro, Valiron 2011]`

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : R}{\Gamma \vdash M + N : T + R} +_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \alpha.T} \alpha_I \qquad \frac{\Gamma \vdash M : T}{\Gamma \vdash \mathbf{0} : 0.T} 0_I$$

Pros:

- Much more versatile
- Good characterisation of terms

Cons:

- Complex
- Overkill (confluence!)

$\boxed{\textbf{Idea: extend } \textit{Additive}}$

Idea: λ^{CA} - Extend additive

Key idea

$$\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \lfloor \alpha \rfloor.T} = \underbrace{T + \cdots + T}_{\lfloor \alpha \rfloor}$$

Idea: λ^{CA} - Extend additive

Key idea

$$\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \lfloor\alpha\rfloor.T} = \underbrace{T + \cdots + T}_{\lfloor\alpha\rfloor}$$

$$\text{If} \vdash M : T, \quad \text{then} \vdash (0.9).M + (1.1).M : T$$

$$(0.9).M + (1.1).M \to 2.M \quad \text{and} \quad \vdash 2.M : 2.T$$

Idea: λ^{CA} - Extend additive

Key idea

$$\frac{\Gamma \vdash M : T}{\Gamma \vdash \alpha.M : \lfloor\alpha\rfloor.T} = \underbrace{T + \cdots + T}_{\lfloor\alpha\rfloor}$$

$$\text{If } \vdash M : T, \quad \text{then } \vdash (0.9).M + (1.1).M : T$$

$$(0.9).M + (1.1).M \to 2.M \quad \text{and} \quad \vdash 2.M : 2.T$$

Theorem (weak subject reduction)

$M \to N, \Gamma \vdash M : T \Rightarrow \Gamma \vdash N : R$ *with* $T \preceq R$

Unicity of types

Second problem:

$$M + M \to 2.M$$

If $\Gamma \vdash M : T$ and $\Gamma \vdash M : R$, $\Gamma \vdash 2.M :?$

Unicity of types

Second problem:

$$M + M \to 2.M$$

If $\Gamma \vdash M : T$ and $\Gamma \vdash M : R$, $\Gamma \vdash 2.M :?$

$\boxed{\text{Solution: Church style}}$

Confluence

$$\boxed{\text{Local confluence} + \text{Strong normalisation} \Rightarrow \text{Confluence}}$$

Confluence

$\boxed{\text{Local confluence + Strong normalisation} \Rightarrow \text{Confluence}}$

- Local confluence: Coq proof.

Confluence

$\boxed{\text{Local confluence} + \text{Strong normalisation} \Rightarrow \text{Confluence}}$

- Local confluence: Coq proof.
- Strong normalisation:
  Plan
  - Translation from λ^{CA} to λ_{lin} (*i.e.* remove annotations)
  - Preservation of reduction by the translation
  - Typability in $\lambda^{\mathrm{CA}} \Rightarrow$ Typability in *Vectorial*
  - SN in *Vectorial* $\Rightarrow$ SN in λ^{CA}

Strong normalisation

$|M|$ is the term M without type annotations

Strong normalisation

$|M|$ is the term M without type annotations

Lemma

If $M \to N$, then

$$|M| \to_v^= |N|$$

where $|M| = |N|$ only when $M \to N$ is a type beta reduction.

Strong normalisation

$|M|$ is the term M without type annotations

Lemma

If $M \to N$, then

$$|M| \to_v^= |N|$$

where $|M| = |N|$ only when $M \to N$ is a type beta reduction.

Lemma

$$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R$$

Strong normalisation

Lemma

There is no infinite sequence reduction consisting only of type beta rules

Strong normalisation

Lemma

There is no infinite sequence reduction consisting only of type beta rules

Proof.

Define:

$$\begin{array}{ccc} \sigma(x:U)=1 & \sigma(\Lambda X.\mathbf{t})=1+\sigma(\mathbf{t}) & \sigma(\mathbf{t}@U)=\sigma(\mathbf{t}) \\ \sigma(\lambda x:U.\mathbf{t})=\sigma(\mathbf{t}) & \sigma((\mathbf{t})\ \mathbf{r})=\sigma(\mathbf{t})\ \sigma(\mathbf{r}) & \sigma(\mathbf{0})=1 \\ \sigma(\alpha.\mathbf{t})=\sigma(\mathbf{t}) & & \sigma(\mathbf{t}+\mathbf{r})=\sigma(\mathbf{t})+\sigma(\mathbf{r}) \end{array}$$

Induction on M: $\sigma((\Lambda X.M)@T) > \sigma(M[X/T])$. □

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M \textit{ is SN}$

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M \textit{ is SN}$

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M \textit{ is SN}$

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN
Assume M not SN:

$$M \rightarrow\ M_1\ \rightarrow\ M_2\ \rightarrow \cdots$$

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M$ *is SN*

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN

Assume M not SN:

$$M \to \; M_1 \; \to \; M_2 \; \to \cdots$$

Then

$$|M| \to_v^= |M_1| \to_v^= |M_2| \to_v^= \cdots$$

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M$ *is SN*

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN
Assume M not SN:

$$M \to \ M_1 \ \to \ M_2 \ \to \cdots$$

Then

$$|M| \to_v^= |M_1| \to_v^= |M_2| \to_v^= \cdots$$

$|M|$ is SN $\Rightarrow \forall i > n \ |M_i| = |M_{i+1}|$

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M \textit{ is SN}$

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN
Assume M not SN:

$$M \to \ M_1 \ \to \ M_2 \ \to \cdots$$

Then

$$|M| \to_v^= |M_1| \to_v^= |M_2| \to_v^= \cdots$$

$|M|$ is SN $\Rightarrow \forall i > n \ |M_i| = |M_{i+1}|$
Then $M_i \to M_{i+1} \to \cdots$ are type beta reductions.

Strong normalisation

Theorem (Strong normalisation)

$\Gamma \vdash M : T \Rightarrow M \textit{ is SN}$

Proof.

$\Gamma \vdash M : T \Rightarrow \Delta \vdash_v |M| : R \Rightarrow |M|$ is SN
Assume M not SN:

$$M \to \; M_1 \; \to \; M_2 \; \to \cdots$$

Then

$$|M| \to_v^= |M_1| \to_v^= |M_2| \to_v^= \cdots$$

$|M|$ is SN $\Rightarrow \forall i > n \; |M_i| = |M_{i+1}|$
Then $M_i \to M_{i+1} \to \cdots$ are type beta reductions.
Then M_i must be SN. Absurd. □

Abstract interpretation

Translation to additive:

$$\tau(\alpha.M) = \sum_{i=1}^{\lfloor \alpha \rfloor} \tau(M)$$

Abstract interpretation

Translation to additive:

$$\tau(\alpha.M) = \sum_{i=1}^{\lfloor \alpha \rfloor} \tau(M)$$

Theorem (Abstract interpretation)

$$\tau(M) \downarrow_a \preceq \tau(M \downarrow)$$

Abstract interpretation

Translation to additive:

$$\tau(\alpha.M) = \sum_{i=1}^{\lfloor\alpha\rfloor} \tau(M)$$

Theorem (Abstract interpretation)

$$\tau(M) \downarrow_a \preceq \tau(M \downarrow)$$

Lemma (Typing preservation)

$$\Gamma \vdash M : T \Rightarrow \Gamma \vdash_a \tau(M) : T$$

Abstract interpretation

$$\begin{tikzcd}
\lambda^{\mathrm{CA}} \arrow[r, "\tau"] \arrow[d, "\downarrow"'] & \lambda_{add} \arrow[r, "{[\cdot]}"] \arrow[d, "\downarrow_a", "(\preceq)"'] & F_p \arrow[d, "\downarrow_F", "(\preceq)"'] \\
\lambda^{\mathrm{CA}} \arrow[r, "\tau"'] & \lambda_{add} \arrow[r, "{[\cdot]}"'] & F_p
\end{tikzcd}$$

Contributions

- "Powerful" alternative to *Vectorial* (extension of *Additive*)
- (weak) Subject reduction
- Strong normalisation (via translation to *Vectorial*)
- Confluence (via SN)
- Abstract interpretation in System F with pairs (via *Additive*)

Possible extensions

- Taking also the ceil to have intervals... [Philippe Jorrand]

$$\vdash \alpha.M : [\lfloor\alpha\rfloor, \lceil\alpha\rceil].T$$

(better approximations)

Possible extensions

- Taking also the ceil to have intervals... [Philippe Jorrand]

$$\vdash \alpha.M : [\lfloor\alpha\rfloor, \lceil\alpha\rceil].T$$

(better approximations)

- pairs to represent negatives... [Simon Perdrix]

$$\vdash \alpha.M : (\lfloor\alpha\rfloor, \star).T \qquad \vdash -\alpha.M : (\star, \lfloor\alpha\rfloor).T$$

Possible extensions

- Taking also the ceil to have intervals... `[Philippe Jorrand]`

$$\vdash \alpha.M : [\lfloor\alpha\rfloor, \lceil\alpha\rceil].T$$

  (better approximations)

- pairs to represent negatives... `[Simon Perdrix]`

$$\vdash \alpha.M : (\lfloor\alpha\rfloor, \star).T \qquad\qquad \vdash -\alpha.M : (\star, \lfloor\alpha\rfloor).T$$

- Complexes... `[Simon Perdrix]`

$$\vdash (\alpha - \beta i).M : [(\lfloor\alpha\rfloor, \star), (\star, \lfloor\beta\rfloor)].T$$

- etc.