Constructing Confluence

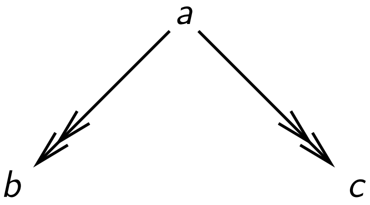
Vincent van Oostrom

Theoretical Philosophy
Utrecht University
The Netherlands

September 8, 2006

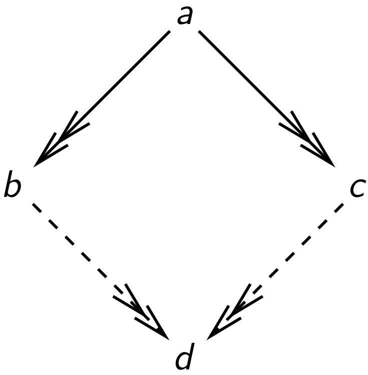
Confluence of rewrite relation $\longrightarrow$

Confluence of rewrite relation $\longrightarrow$



- $\forall a, b, c$ such that a reduces to b,c

Confluence of rewrite relation $\longrightarrow$



- $\forall a, b, c$ such that a reduces to b,c
- $\exists d$ such that b, c reduce to d

Relations vs. systems

Rewrite relation?

Relations vs. systems

No, want to construct valley on basis of steps in peak

Relations vs. systems

Rewrite system!

Relations vs. systems

- Definition
  Abstract Rewriting System is $\langle A, \Phi, \text{src}, \text{tgt} \rangle$
    - A set of objects
    - Φ set of steps
    - src, tgt : $\Phi \longrightarrow A$
      source, target functions

Relations vs. systems

- Definition
  Abstract Rewriting System is $\langle A, \Phi, \mathrm{src}, \mathrm{tgt} \rangle$
    - A set of objects
    - Φ set of steps
    - src, tgt : $\Phi \to A$
      source, target functions

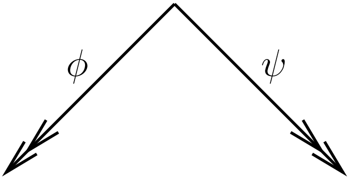
- Steps ϕ, ψ, χ, ω, $\ldots$

Relations vs. systems

- Definition
  Abstract Rewriting System is $\langle A, \Phi, \text{src}, \text{tgt}\rangle$
    - A set of objects
    - Φ set of steps
    - src, tgt : $\Phi \rightarrow A$
      source, target functions
- Steps ϕ, ψ, χ, ω, ...
- ϕ : $a \rightarrow b$
  ϕ is step with source a and target b

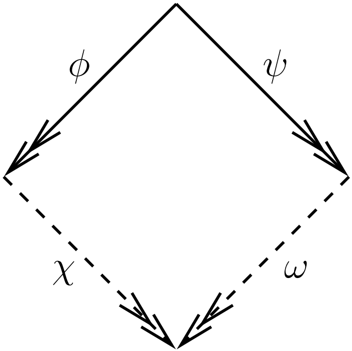
Confluence of rewrite system $\longrightarrow$

Confluence of rewrite system $\longrightarrow$



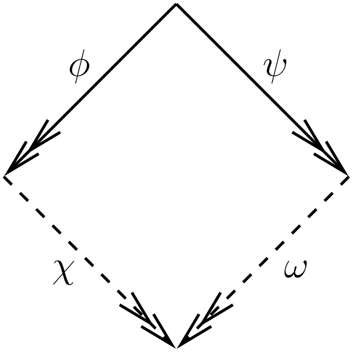
- $\forall \phi, \psi$ co-initial reductions (peak)

Confluence of rewrite system $\longrightarrow$



- $\forall \phi, \psi$ co-initial reductions (peak)
- $\exists \chi, \omega$ co-final reductions (valley)

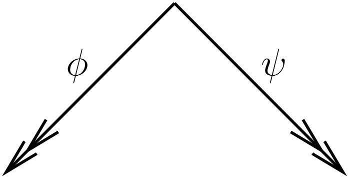
Confluence of rewrite system $\longrightarrow$



- $\forall \phi, \psi$ co-initial reductions (peak)
- $\exists \chi, \omega$ co-final reductions (valley)
- tgt(ϕ) = src(χ), tgt(ψ) = src(ω) (diagram)

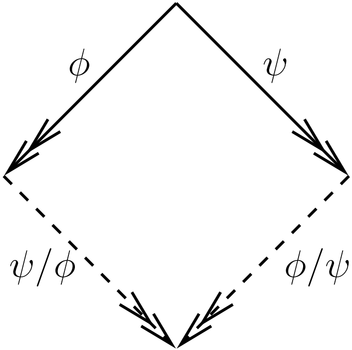
Constructive confluence

Constructive confluence



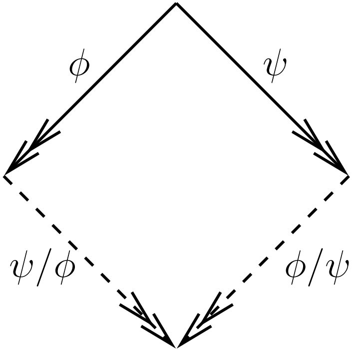
- $\forall \phi, \psi$ peak

Constructive confluence



- $\forall \phi, \psi$ peak
- ψ/ϕ, ϕ/ψ construct valley

Constructive confluence



- $\forall \phi, \psi$ peak
- ψ/ϕ, ϕ/ψ construct valley
- $\mathsf{tgt}(\phi) = \mathsf{src}(\psi/\phi)$, $\mathsf{tgt}(\psi) = \mathsf{src}(\phi/\psi)$ (diagram)

Residual function

- / residual function

Residual function

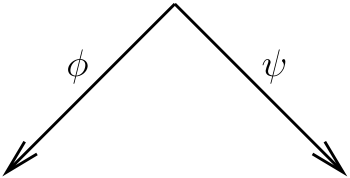
- $/$ residual function
- witnessing constructive confluence proof

Residual function

- / residual function
- witnessing constructive confluence proof
- from peaks to valleys constructing diagrams

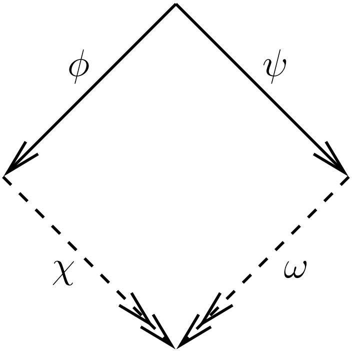
Confluence by Local Confluence?

Confluence by Local Confluence?



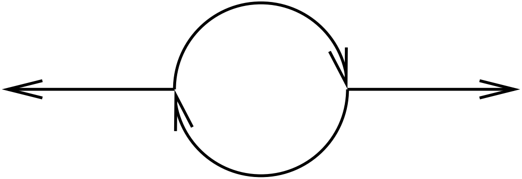
- $\forall \phi, \psi$ co-initial steps (local peak)

Confluence by Local Confluence?



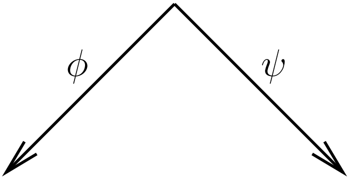
- $\forall \phi, \psi$ co-initial steps (local peak)
- $\exists \chi, \omega$ co-final reductions (valley)

Confluence by Local Confluence?



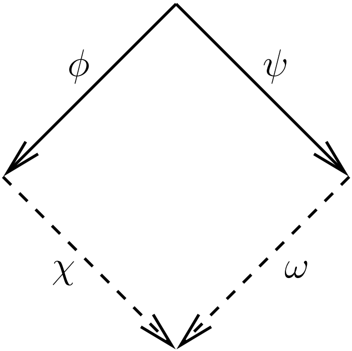
- No confluence (Counterexample Kleene)

Confluence by Local Confluence?



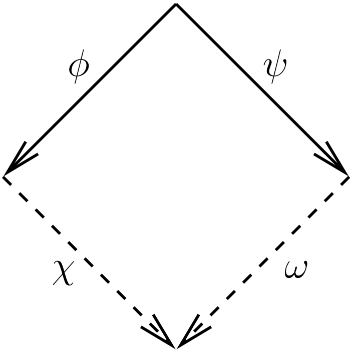
- $\forall \phi, \psi$ co-initial steps (local peak)

Confluence by Local Confluence?



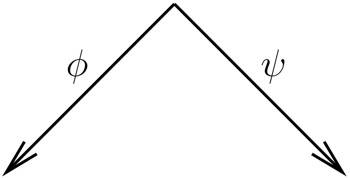
- $\forall \phi, \psi$ co-initial steps (local peak)
- $\exists \chi, \omega$ co-final steps (local valley)

Confluence by Local Confluence?



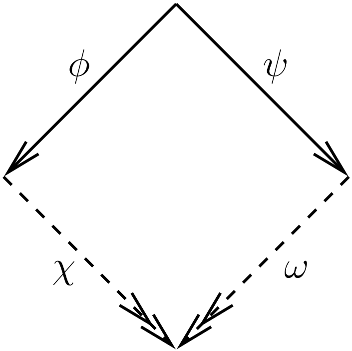
- $\forall \phi, \psi$ co-initial steps (local peak)
- $\exists \chi, \omega$ co-final steps (local valley)
- Diamond property $\Rightarrow$ confluence (Newman)

Confluence by Local Confluence?



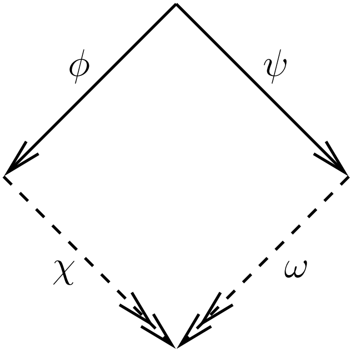
- $\forall \phi, \psi$ co-initial steps (local peak)

Confluence by Local Confluence?



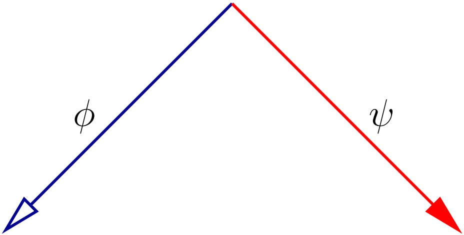
- $\forall \phi, \psi$ co-initial steps (local peak)
- $\exists \chi, \omega$ co-final reductions (valley) & $\rightarrow$ is terminating

Confluence by Local Confluence?



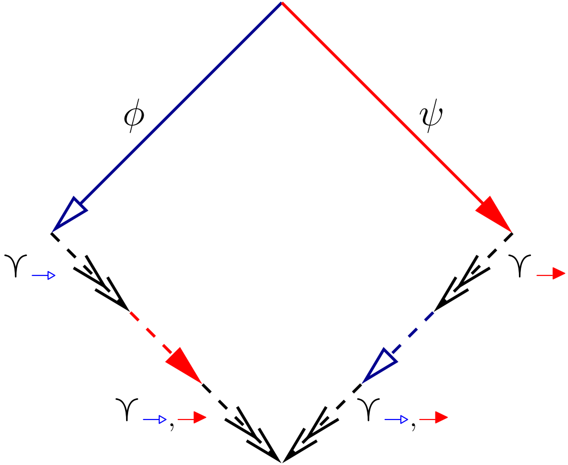
- $\forall \phi, \psi$ co-initial steps (local peak)
- $\exists \chi, \omega$ co-final reductions (valley) & $\rightarrow$ is terminating
- Local confluence & termination $\Rightarrow$ confluence (Newman)

Confluence by Local Confluence?



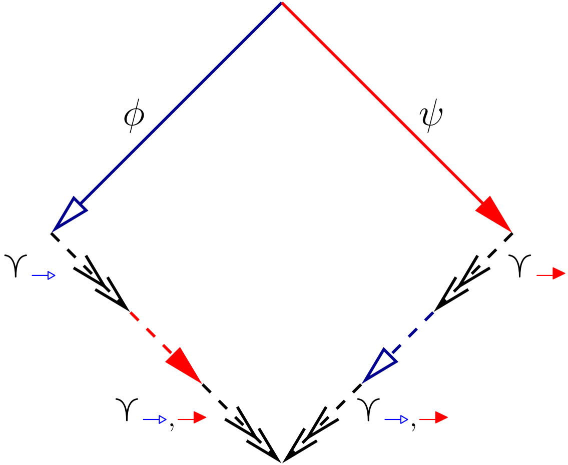
- $\forall$ ⇾, → $\in A$, ⇾-step ϕ, →-step ψ, co-initial

Confluence by Local Confluence?



- $\forall \rightarrowtriangle, \rightarrow \in A$, $\rightarrowtriangle$-step ϕ, $\rightarrow$-step ψ, co-initial
- $\exists$ decreasing co-final reductions for well-founded order $(A, \prec)$

Confluence by Local Confluence?



- $\forall$ ⇾, → $\in A$, ⇾-step ϕ, →-step ψ, co-initial
- $\exists$ decreasing co-final reductions for well-founded order $(A, \prec)$
- Decreasing diagrams $\Rightarrow$ confluence of $\bigcup A$ (vO)

Decreasing diagrams method

- given rewrite system $\longrightarrow$

Decreasing diagrams method

- given rewrite system $\rightarrow$
- decompose $\rightarrow$ into set A of rewrite systems ($\rightarrow = \bigcup A$)

Decreasing diagrams method

- given rewrite system $\rightarrow$
- decompose $\rightarrow$ into set A of rewrite systems ($\rightarrow = \bigcup A$)
- well-foundedly ordered ($\prec$)

Decreasing diagrams method

- given rewrite system $\to$
- decompose $\to$ into set A of rewrite systems ($\to = \bigcup A$)
- well-foundedly ordered ($\prec$)
- $\forall$ co-initial $\to$ and $\to$ steps

Decreasing diagrams method

- given rewrite system $\rightarrow$
- decompose $\rightarrow$ into set A of rewrite systems $(\rightarrow = \bigcup A)$
- well-foundedly ordered $(\prec)$
- $\forall$ co-initial $\color{blue}{\rightarrow}$ and $\color{red}{\rightarrow}$ steps
- $\exists$ co-final $\color{blue}{\rightarrow}$,$\color{red}{\rightarrow}$-decreasing and $\color{red}{\rightarrow}$,$\color{blue}{\rightarrow}$-decreasing reductions

Decreasing diagrams method

- given rewrite system $\rightarrow$
- decompose $\rightarrow$ into set A of rewrite systems $(\rightarrow = \bigcup A)$
- well-foundedly ordered $(\prec)$
- $\forall$ co-initial $\textcolor{blue}{\rightarrow}$ and $\textcolor{red}{\rightarrow}$ steps
- $\exists$ co-final $\textcolor{blue}{\rightarrow}$,$\textcolor{red}{\rightarrow}$-decreasing and $\textcolor{red}{\rightarrow}$,$\textcolor{blue}{\rightarrow}$-decreasing reductions
- $\textcolor{blue}{\rightarrow}$,$\textcolor{red}{\rightarrow}$-decreasing: steps below $\textcolor{blue}{\rightarrow}$; $\textcolor{blue}{\rightarrow}$-step; steps below $\textcolor{blue}{\rightarrow}$, $\textcolor{red}{\rightarrow}$

Decreasing diagrams method

- given rewrite system $\to$
- decompose $\to$ into set A of rewrite systems ($\to = \bigcup A$)
- well-foundedly ordered ($\prec$)
- $\forall$ co-initial $\rightarrowtriangle$ and $\rightarrow$ steps
- $\exists$ co-final $\rightarrowtriangle$,$\rightarrow$-decreasing and $\rightarrow$,$\rightarrowtriangle$-decreasing reductions
- $\rightarrowtriangle$,$\rightarrow$-decreasing: steps below $\rightarrowtriangle$; $\rightarrowtriangle$-step; steps below $\rightarrowtriangle$, $\rightarrow$
- constructive (tiling)

Decreasing diagrams method

- given rewrite system $\to$
- decompose $\to$ into set A of rewrite systems ($\to = \bigcup A$)
- well-foundedly ordered ($\prec$)
- $\forall$ co-initial $\color{blue}{\rightarrowtriangle}$ and $\color{red}{\blacktriangleright}$ steps
- $\exists$ co-final $\color{blue}{\rightarrowtriangle}$,$\color{red}{\blacktriangleright}$-decreasing and $\color{red}{\blacktriangleright}$,$\color{blue}{\rightarrowtriangle}$-decreasing reductions
- $\color{blue}{\rightarrowtriangle}$,$\color{red}{\blacktriangleright}$-decreasing: steps below $\color{blue}{\rightarrowtriangle}$; $\color{blue}{\rightarrowtriangle}$-step; steps below $\color{blue}{\rightarrowtriangle}$, $\color{red}{\blacktriangleright}$

- constructive (tiling)
- graph rewriting (Blom), explicit substitutions (vO)

Decreasing diagrams method

- given rewrite system $\to$
- decompose $\to$ into set A of rewrite systems ($\to = \bigcup A$)
- well-foundedly ordered ($\prec$)
- $\forall$ co-initial $\textcolor{blue}{\rightarrowtriangle}$ and $\textcolor{red}{\rightarrow}$ steps
- $\exists$ co-final $\textcolor{blue}{\rightarrowtriangle}$,$\textcolor{red}{\rightarrow}$-decreasing and $\textcolor{red}{\rightarrow}$,$\textcolor{blue}{\rightarrowtriangle}$-decreasing reductions
- $\textcolor{blue}{\rightarrowtriangle}$,$\textcolor{red}{\rightarrow}$-decreasing: steps below $\textcolor{blue}{\rightarrowtriangle}$; $\textcolor{blue}{\rightarrowtriangle}$-step; steps below $\textcolor{blue}{\rightarrowtriangle}$, $\textcolor{red}{\rightarrow}$

- constructive (tiling)
- graph rewriting (Blom), explicit substitutions (vO)
- ambients (Lévy), bisimilarity (Pous), modularity (vO), ...

Decreasing diagrams method

- given rewrite system $\rightarrow$
- decompose $\rightarrow$ into set A of rewrite systems ($\rightarrow = \bigcup A$)
- well-foundedly ordered ($\prec$)
- $\forall$ co-initial $\color{blue}{\rightarrow}$ and $\color{red}{\rightarrow}$ steps
- $\exists$ co-final $\color{blue}{\rightarrow}$,$\color{red}{\rightarrow}$-decreasing and $\color{red}{\rightarrow}$,$\color{blue}{\rightarrow}$-decreasing reductions
- $\color{blue}{\rightarrow}$,$\color{red}{\rightarrow}$-decreasing: steps below $\color{blue}{\rightarrow}$; $\color{blue}{\rightarrow}$-step; steps below $\color{blue}{\rightarrow}$, $\color{red}{\rightarrow}$

- constructive (tiling)
- graph rewriting (Blom), explicit substitutions (vO)
- ambients (Lévy), bisimilarity (Pous), modularity (vO), ...
- complete for countable rewrite systems (open otherwise)

Confluence of Combinatory Logic?

$$\begin{aligned} A(I,x) &= x \\ A(A(K,x),y) &= x \\ A(A(A(S,x),y),z) &= A(A(x,z),A(y,z)) \end{aligned}$$

- Combinatory equational logic (Schönfinkel, Curry)

Confluence of Combinatory Logic?

$$
\begin{aligned}
A(I, x) &= x \\
A(A(K, x), y) &= x \\
A(A(A(S, x), y), z) &= A(A(x, z), A(y, z))
\end{aligned}
$$

- Combinatory equational logic (Schönfinkel, Curry)
- A application, I identity, K constant, S substitution

Confluence of Combinatory Logic?

$$
\begin{aligned}
(I \cdot x) &= x \\
((K \cdot x) \cdot y) &= x \\
(((S \cdot x) \cdot y) \cdot z) &= ((x \cdot z) \cdot (y \cdot z))
\end{aligned}
$$

- $\cdot$ infix application

Confluence of Combinatory Logic?

$$
\begin{aligned}
I \cdot x &= x \\
K \cdot x \cdot y &= x \\
S \cdot x \cdot y \cdot z &= x \cdot z \cdot (y \cdot z)
\end{aligned}
$$

- $\cdot$ left-associative

Confluence of Combinatory Logic?

$$
\begin{aligned}
Ix &= x \\
Kxy &= x \\
Sxyz &= xz(yz)
\end{aligned}
$$

- $\cdot$ denoted by juxtaposition

Confluence of Combinatory Logic?

$$
\begin{aligned}
Ix &\rightarrow x \\
Kxy &\rightarrow x \\
Sxyz &\rightarrow xz(yz)
\end{aligned}
$$

- Combinatory rewriting logic (CL)

Confluence of Combinatory Logic?

$$
\begin{aligned}
Ix &\to x \\
Kxy &\to x \\
Sxyz &\to xz(yz)
\end{aligned}
$$

- Combinatory rewriting logic (CL)
- CL constructively confluent?

Combinatory equational logic

$$\frac{}{l = r}\,(l = r) \quad \frac{s = t}{s^\tau = t^\tau}\,(\text{substitutive}) \quad \frac{}{c = c}\,(c) \quad \frac{s_1 = t_1 \quad s_2 = t_2}{s_1 s_2 = t_1 t_2}\,()$$

$$\frac{}{s = s}\,(\text{reflexive}) \quad \frac{s = t}{t = s}\,(\text{symmetric}) \quad \frac{s = t \quad t = u}{s = u}\,(\text{transitive})$$

Combinatory equational logic

$$\frac{}{l = r}\,(l = r) \quad \frac{s = t}{s^{\tau} = t^{\tau}}\,(\text{substitutive}) \quad \frac{}{c = c}\,(c) \quad \frac{s_1 = t_1 \quad s_2 = t_2}{s_1 s_2 = t_1 t_2}\,()$$

$$\frac{}{s = s}\,(\text{reflexive}) \quad \frac{s = t}{t = s}\,(\text{symmetric}) \quad \frac{s = t \quad t = u}{s = u}\,(\text{transitive})$$

Theorem

$t \approx s \iff t \leftrightarrow^* s \iff t = s$ *(Birkhoff)*

Combinatory equational logic

$$\frac{}{l = r}\,(l = r) \quad \frac{s = t}{s^\tau = t^\tau}\,(\text{substitutive}) \quad \frac{}{c = c}\,(c) \quad \frac{s_1 = t_1 \quad s_2 = t_2}{s_1 s_2 = t_1 t_2}\,()$$

$$\frac{}{s = s}\,(\text{reflexive}) \quad \frac{s = t}{t = s}\,(\text{symmetric}) \quad \frac{s = t \quad t = u}{s = u}\,(\text{transitive})$$

Rewriting logic = Equational logic − symmetry (Meseguer)

Combinatory rewriting logic

$$\frac{}{l \geq r}(l \to r) \quad \frac{s \geq t}{s^{\tau} \geq t^{\tau}}(\text{substitutive}) \quad \frac{}{c \geq c}(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}()$$

$$\frac{}{s \geq s}(\text{reflexive}) \qquad \frac{s \geq t \quad t \geq u}{s \geq u}(\text{transitive})$$

Combinatory rewriting logic

$$\frac{}{l \geq r}(l \to r) \quad \frac{s \geq t}{s^\tau \geq t^\tau}(\text{substitutive}) \quad \frac{}{c \geq c}(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}()$$

$$\frac{}{s \geq s}(\text{reflexive}) \qquad \frac{s \geq t \quad t \geq u}{s \geq u}(\text{transitive})$$

Theorem

$t \succeq s \iff t \twoheadrightarrow s \iff t \geq s$ (*vO*)

Combinatory rewriting logic

$$\frac{}{l \geq r}\,(l \to r) \quad \frac{s \geq t}{s^{\tau} \geq t^{\tau}}\,(\text{substitutive}) \quad \frac{}{c \geq c}\,(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}\,()$$

$$\frac{}{s \geq s}\,(\text{reflexive}) \qquad \frac{s \geq t \quad t \geq u}{s \geq u}\,(\text{transitive})$$

On closed terms reflexivity superfluous (use congruence)

Combinatory rewriting logic

$$\frac{}{l \geq r}(l \to r) \quad \frac{s \geq t}{s^\tau \geq t^\tau}(\text{substitutive}) \quad \frac{}{c \geq c}(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}(\text{transitive})$$

On closed terms reflexivity superfluous (use congruence)

Combinatory rewriting logic

$$\frac{}{l \geq r}\,(l \to r) \quad \frac{s \geq t}{s^\tau \geq t^\tau}\,(\text{substitutive}) \quad \frac{}{c \geq c}\,(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}\,()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}\,(\text{transitive})$$

Allow rewriting inside substitutions

Combinatory rewriting logic

$$\frac{}{l \geq r}(l \to r) \quad \frac{s \geq t \quad \tau \geq \theta}{s^{\tau} \geq t^{\theta}}(\text{subst.}) \quad \frac{}{c \geq c}(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}(\text{transitive})$$

Allow rewriting inside substitutions

Combinatory rewriting logic

$$\frac{}{l \geq r}\,(l \to r) \quad \frac{s \geq t \quad \tau \geq \theta}{s^{\tau} \geq t^{\theta}}\,(\text{subst.}) \quad \frac{}{c \geq c}\,(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}\,()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}\,(\text{transitive})$$

Substitutivity superfluous (instantiate rules immediately)

Combinatory rewriting logic

$$\frac{\tau \geq \theta}{l^\tau \geq r^\theta}\,(l \to r) \quad \frac{}{c \geq c}\,(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}\,()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}\,(\text{transitive})$$

Combinatory rewriting logic

$$\frac{\tau \geq \theta}{l^{\tau} \geq r^{\theta}} (l \to r) \quad \frac{}{c \geq c} (c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2} ()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u} (\mathrm{transitive})$$

Residual function on combinatory rewriting logic proofs?

Combinatory rewriting logic

$$\frac{\tau \geq \theta}{l^{\tau} \geq r^{\theta}}\,(l \to r) \quad \frac{}{c \geq c}\,(c) \quad \frac{s_1 \geq t_1 \quad s_2 \geq t_2}{s_1 s_2 \geq t_1 t_2}\,()$$

$$\frac{s \geq t \quad t \geq u}{s \geq u}\,(\text{transitive})$$

Residual function on combinatory rewriting logic proof terms!

Combinatory rewriting logic proof terms

$$\begin{aligned} \iota(x) : Ix &\to x \\ \kappa(x,y) : Kxy &\to x \\ \sigma(x,y,z) : Sxyz &\to xz(yz) \end{aligned}$$

$$\frac{\Phi : \tau \geq \theta}{\varrho^{\Phi} : l^{\tau} \geq r^{\theta}} \, (\varrho : l \to r)$$

Combinatory rewriting logic proof terms

$$\begin{array}{rcl} \iota(x) : Ix & \to & x \\ \kappa(x,y) : Kxy & \to & x \\ \sigma(x,y,z) : Sxyz & \to & xz(yz) \end{array}$$

$$\frac{\Phi : \tau \geq \theta}{\varrho^{\Phi} : l^{\tau} \geq r^{\theta}}\,(\varrho : l \to r)$$

$$\frac{}{c : c = c}\,(c) \qquad \frac{\phi : s_1 \geq t_1 \quad \psi : s_2 \geq t_2}{\phi\psi : s_1 s_2 \geq t_1 t_2}\,()$$

Combinatory rewriting logic proof terms

$$\begin{array}{rcl} \iota(x) : \; Ix & \to & x \\ \kappa(x,y) : \; Kxy & \to & x \\ \sigma(x,y,z) : \; Sxyz & \to & xz(yz) \end{array}$$

$$\frac{\Phi : \; \tau \geq \theta}{\varrho^{\Phi} : \; l^{\tau} \geq r^{\theta}} \, (\varrho : \; l \to r)$$

$$\frac{}{c : \; c = c} \, (c) \qquad \frac{\phi : \; s_1 \geq t_1 \quad \psi : \; s_2 \geq t_2}{\phi\psi : \; s_1 s_2 \geq t_1 t_2} \, ()$$

$$\frac{\phi : \; s \geq t \quad \psi : \; t \geq u}{\phi \circ \psi : \; s \geq u} \, (\circ)$$

Combinatory rewriting logic proof term examples

$I(It) \geq It$?

Combinatory rewriting logic proof term examples

$I(It) \geq It$?

$$\dfrac{\dfrac{}{Ix \geq x}\,(Ix \to x)}{I(It) \geq It}\,(\text{substitutive})$$

Combinatory rewriting logic proof term examples

$I(It) \geq It$?

$$\dfrac{\overline{Ix \geq x}\,(Ix \to x)}{I(It) \geq It}\,(\text{substitutive})$$

- $\iota(It) : \; I(It) \geq It$

Combinatory rewriting logic proof term examples

$I(It) \geq It$?

$$\dfrac{\dfrac{}{I \geq I}\,(\text{reflexive}) \quad \dfrac{\dfrac{}{Ix \geq x}\,(Ix \to x)}{It \geq t}\,(\text{substitutive})}{I(It) \geq It}\,()$$

Combinatory rewriting logic proof term examples

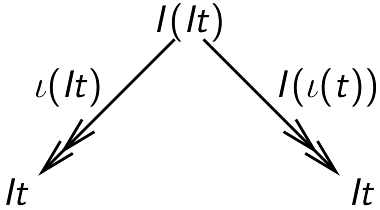
$I(It) \geq It$?

$$\dfrac{\dfrac{}{I \geq I}\,(\text{reflexive}) \qquad \dfrac{\dfrac{}{Ix \geq x}\,(Ix \to x)}{It \geq t}\,(\text{substitutive})}{I(It) \geq It}\,()$$

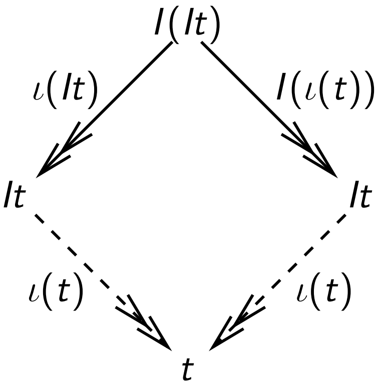
- $I(\iota(t)) : \; I(It) \geq It$

Confluence of combinatory rewriting logic example

Confluence of combinatory rewriting logic example



Confluence of combinatory rewriting logic example



- $I(\iota(t))/\iota(It) = \iota(t) = \iota(It)/I(\iota(t))$

Residual function for combinatory rewriting logic

$$\begin{aligned}
c/c &= c \\
(\phi_1\phi_2)/(\psi_1\psi_2) &= (\phi_1/\psi_1)(\phi_2/\psi_2) \\
\varrho(\phi_1,\ldots,\phi_n)/l(\psi_1,\ldots,\psi_n) &= \varrho(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
l(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\varrho(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\chi/(\phi\circ\psi) &= (\chi/\phi)/\psi \\
(\phi\circ\psi)/\chi &= \phi/\chi\circ\psi/(\chi/\phi)
\end{aligned}$$

Residual function for combinatory rewriting logic

$$\begin{array}{rcl} c/c & = & c \\ (\phi_1\phi_2)/(\psi_1\psi_2) & = & (\phi_1/\psi_1)(\phi_2/\psi_2) \\ \varrho(\phi_1,\ldots,\phi_n)/l(\psi_1,\ldots,\psi_n) & = & \varrho(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\ l(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) & = & r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\ \varrho(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) & = & r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\ \chi/(\phi\circ\psi) & = & (\chi/\phi)/\psi \\ (\phi\circ\psi)/\chi & = & \phi/\chi\circ\psi/(\chi/\phi) \end{array}$$

- Reified inductive confluence proof for orthogonal systems

Residual function for combinatory rewriting logic

$$
\begin{aligned}
c/c &= c \\
(\phi_1\phi_2)/(\psi_1\psi_2) &= (\phi_1/\psi_1)(\phi_2/\psi_2) \\
\varrho(\phi_1,\ldots,\phi_n)/l(\psi_1,\ldots,\psi_n) &= \varrho(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
l(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\varrho(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\chi/(\phi\circ\psi) &= (\chi/\phi)/\psi \\
(\phi\circ\psi)/\chi &= \phi/\chi\circ\psi/(\chi/\phi)
\end{aligned}
$$

- Reified inductive confluence proof for orthogonal systems
- OTRSs (Rosen), $\lambda\beta$ (Tait & Martin-Löf), HOTRS (vO)

Residual function for combinatory rewriting logic

$$
\begin{aligned}
c/c &= c \\
(\phi_1\phi_2)/(\psi_1\psi_2) &= (\phi_1/\psi_1)(\phi_2/\psi_2) \\
\varrho(\phi_1,\ldots,\phi_n)/l(\psi_1,\ldots,\psi_n) &= \varrho(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
l(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\varrho(\phi_1,\ldots,\phi_n)/\varrho(\psi_1,\ldots,\psi_n) &= r(\phi_1/\psi_1,\ldots,\phi_n/\psi_n) \\
\chi/(\phi\circ\psi) &= (\chi/\phi)/\psi \\
(\phi\circ\psi)/\chi &= \phi/\chi\circ\psi/(\chi/\phi)
\end{aligned}
$$

- Reified inductive confluence proof for orthogonal systems
- OTRSs (Rosen), $\lambda\beta$ (Tait & Martin-Löf), HOTRS (vO)
- Characterize orthogonality abstractly via / ?

Residual systems

Definition

Residual system is $\langle \rightarrow, 1, /, \circ \rangle$

- $\rightarrow$ an abstract rewriting system
- $/$ residual function
- 1 unit function $\mathsf{tgt}(1_a) = a = \mathsf{src}(1_a)$
- $\circ$ composition function on ϕ, ψ s.t. $\mathsf{tgt}(\phi) = \mathsf{src}(\psi)$

$$
\begin{array}{rcl}
\phi/\phi & = & 1 \\
\phi/1 & = & \phi \\
1/\phi & = & 1 \\
(\phi/\psi)/(\chi/\psi) & = & (\phi/\chi)/(\psi/\chi) \\
1 \circ 1 & = & 1 \\
\chi/(\phi \circ \psi) & = & (\chi/\phi)/\psi \\
(\phi \circ \psi)/\chi & = & (\phi/\chi) \circ (\psi/(\chi/\phi))
\end{array}
$$

Natural numbers as residual algebra

- Objects: $\{*\}$ (single object)
- Steps: $\mathbb{N}$ (natural numbers)
- Residual: $\dot{-}$ (cut-off subtraction)
- Unit: 0 (zero)
- Composition: $+$ (addition)

$$
\begin{aligned}
n \dot{-} n &= 0 \\
n \dot{-} 0 &= n \\
0 \dot{-} n &= 0 \\
(n \dot{-} m) \dot{-} (k \dot{-} m) &= (n \dot{-} k) \dot{-} (m \dot{-} k) \\
0 + 0 &= 0 \\
k \dot{-} (n + m) &= (k \dot{-} n) \dot{-} m \\
(n + m) \dot{-} k &= (n \dot{-} k) + (m \dot{-} (k \dot{-} n))
\end{aligned}
$$

Multisets as residual algebra

- Objects: $\{*\}$ (single object)
- Steps: $\mathbf{Mst}(A)$ (multisets over A)
- Residual: $-$ (multiset difference)
- Unit: $\emptyset$ (empty multiset)
- Composition: $\uplus$ (multiset sum)

$$
\begin{aligned}
M - M &= \emptyset \\
M - \emptyset &= M \\
\emptyset - M &= \emptyset \\
(M - N) - (K - N) &= (M - K) - (N - K) \\
\emptyset \uplus \emptyset &= \emptyset \\
K - (M \uplus N) &= (K - M) - N \\
(M \uplus N) - K &= (M - K) \uplus (N - (K - M))
\end{aligned}
$$

Commutative residual algebras

Definition

commutative residual algebra also satisfies

$$\begin{aligned}(\phi/\psi)/\phi &= 1\\ \phi/(\phi/\psi) &= \psi/(\psi/\phi)\end{aligned}$$

Commutative residual algebras

Definition

commutative residual algebra also satisfies

$$\begin{aligned} (\phi/\psi)/\phi &= 1 \\ \phi/(\phi/\psi) &= \psi/(\psi/\phi) \end{aligned}$$

residual and composition are total (algebra)

Commutative residual algebras

Definition

commutative residual algebra also satisfies

$$\begin{aligned}(\phi/\psi)/\phi &= 1 \\ \phi/(\phi/\psi) &= \psi/(\psi/\phi)\end{aligned}$$

example: natural numbers with cut-off division

Commutative residual algebras

Definition

commutative residual algebra also satisfies

$$\begin{aligned}(\phi/\psi)/\phi &= 1 \\ \phi/(\phi/\psi) &= \psi/(\psi/\phi)\end{aligned}$$

$\preceq$ well-founded ($a \preceq b$ if $a/b = 1$) $\Rightarrow$ multisets (Visser,vO)

Commutative residual algebras

Definition

commutative residual algebra also satisfies

$$\begin{array}{rcl} (\phi/\psi)/\phi & = & 1 \\ \phi/(\phi/\psi) & = & \psi/(\psi/\phi) \end{array}$$

$\preceq$ well-founded ($a \preceq b$ if $a/b = 1$) $\Rightarrow$ multisets (Visser,vO)
iso to commutative BCK algebras with relative cancellation

Commutative residual algebras

Definition

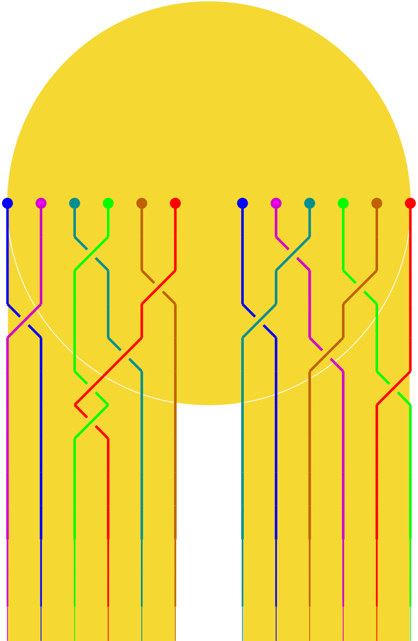
commutative residual algebra also satisfies

$$\begin{aligned}(\phi/\psi)/\phi &= 1 \\ \phi/(\phi/\psi) &= \psi/(\psi/\phi)\end{aligned}$$

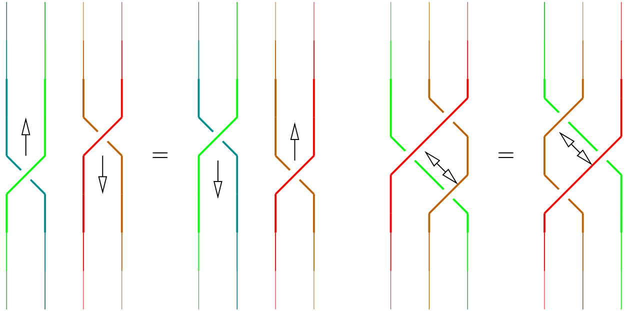
$\preceq$ well-founded ($a \preceq b$ if $a/b = 1$) $\Rightarrow$ multisets (Visser,vO)
Uniquely decomposes into atoms (Luttik,vO)

- $\preceq$ well-founded partial-order
- 1 least
- strictly compatible: $\phi \prec \psi \;\Rightarrow\; \phi \circ \chi \prec \psi \circ \chi$
- precompositional: $\phi \preceq \psi \circ \chi \;\Rightarrow\; \phi = \psi' \circ \chi'$, $\psi' \preceq \psi$, $\chi' \preceq \chi$
- Archimedean: $\forall n\; \phi^n \preceq \psi \;\Rightarrow\; \phi = 1$.

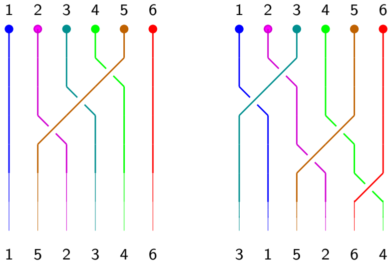
Braid problem



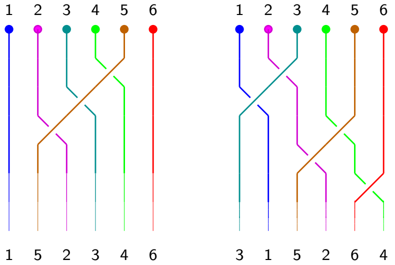
Braid identities



Braids as residual system

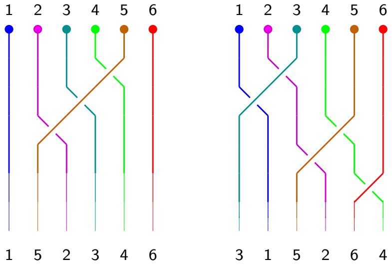


Braids as residual system



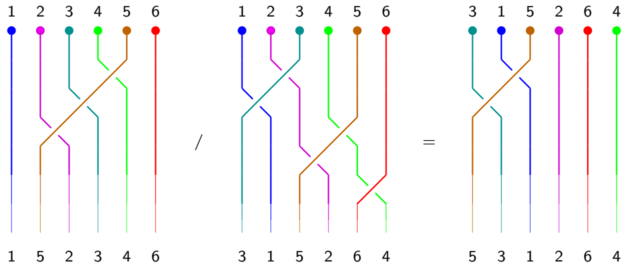
- Objects: relations on strands
  total ($i\ R\ j$ or $j\ R\ i$), irreflexive ($\neg(iRi)$), transitive ($R^+ = R$)

Braids as residual system



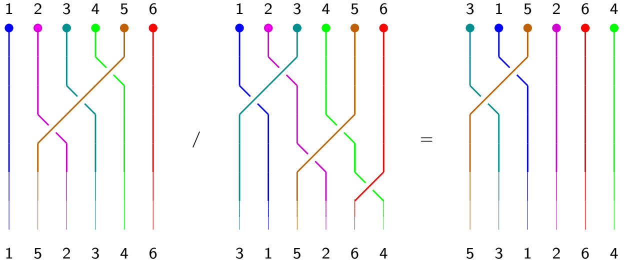
- Objects: relations on strands
- Steps: sequences of multi-steps
  $R - S : R \rightarrow S$ (fastest way from one state to another)
  $\langle 2,5\rangle, \langle 3,5\rangle, \langle 4,5\rangle$ and $\langle 1,3\rangle, \langle 2,3\rangle, \langle 2,5\rangle, \langle 4,5\rangle, \langle 4,6\rangle$

Braids as residual system



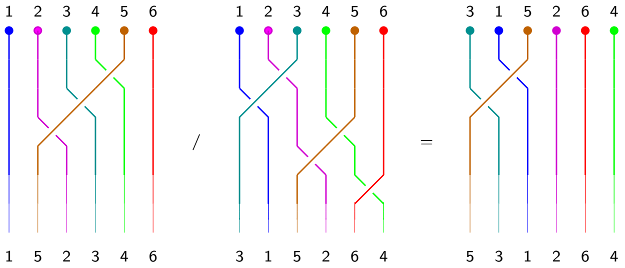
- Objects: relations on strands
- Steps: sequences of multi-steps
  $R - S : R \rightarrow S$ (fastest way from one state to another)
- Residual: $\psi/\phi = (\phi \cup \psi)^{+} - \phi$

Braids as residual system



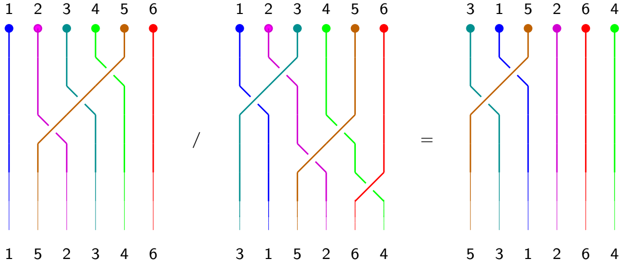
- Objects: relations on strands
- Steps: sequences of multi-steps
  $R - S : R \rightarrow S$ (fastest way from one state to another)
- Residual: $\psi/\phi = (\phi \cup \psi)^+ - \phi$
- Unit: $\emptyset$

Braids as residual system



- Objects: relations on strands
- Steps: sequences of multi-steps
  $R - S : R \to S$ (fastest way from one state to another)
- Residual: $\psi/\phi = (\phi \cup \psi)^+ - \phi$
- Unit: $\emptyset$
- Composition: concatenation

Braids as residual system



- Objects: relations on strands
- Steps: sequences of multi-steps
  $R - S : R \rightarrow S$ (fastest way from one state to another)
- Residual: $\psi/\phi = (\phi \cup \psi)^{+} - \phi$
- Unit: $\emptyset$
- Composition: concatenation

Theorem

Braids constitute residual system (Klop, vO, de Vrijer)

Self-distributivity

$$(x \cdot y) \cdot z = (x \cdot z) \cdot (y \cdot z)$$

Self-distributivity

$$xyz \longrightarrow xz(yz) \text{ (} S\text{-less } S\text{-rule)}$$

Self-distributivity

$$xyz \rightarrow xz(yz) \text{ (S-less S-rule)}$$

- Equational theory (Dehornoy)

Self-distributivity

$$xyz \to xz(yz) \text{ (}S\text{-less } S\text{-rule)}$$

- Equational theory (Dehornoy)
- Residual system (Arbiser, vO)

$xyz \longrightarrow xz(yz)$

Interpret as first projection

$xyz \longrightarrow xz(yz)$

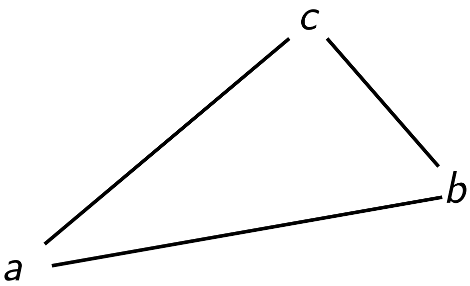
Interpret as an ACI-operation

$$\begin{aligned}
(x \cdot y) \cdot z &= x \cdot (y \cdot z) \\
&= x \cdot (y \cdot (z \cdot z)) \\
&= x \cdot ((y \cdot z) \cdot z) \\
&= x \cdot (z \cdot (y \cdot z)) \\
&= (x \cdot z) \cdot (y \cdot z)
\end{aligned}$$

Examples: disjunction/union, conjunction/intersection

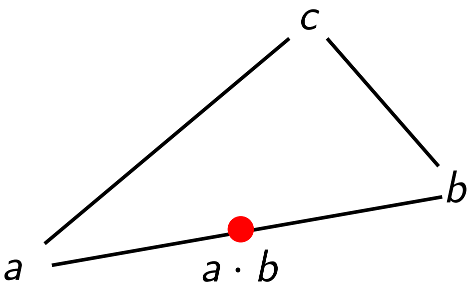
$xyz \longrightarrow xz(yz)$

Interpret as 'middle'



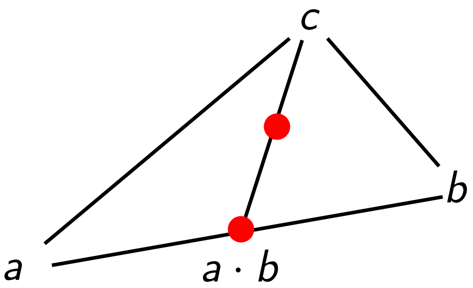
$xyz \to xz(yz)$

Interpret as 'middle'



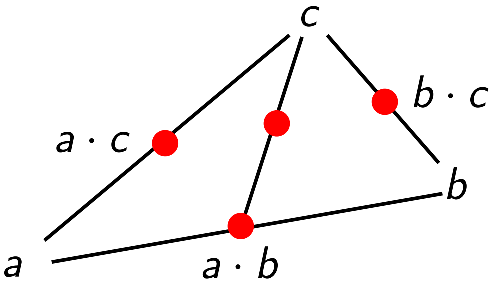
$xyz \longrightarrow xz(yz)$

Interpret as 'middle'



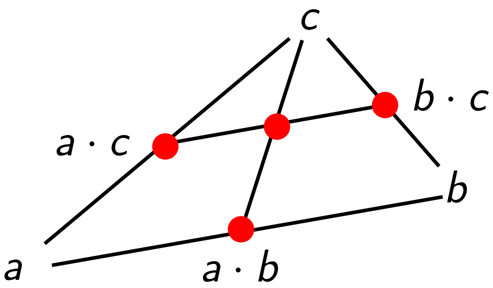
$xyz \longrightarrow xz(yz)$

Interpret as 'middle'



$xyz \to xz(yz)$

Interpret as ‘middle’

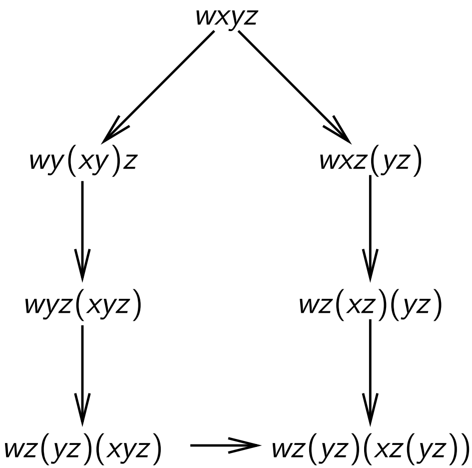


$xyz \to xz(yz)$

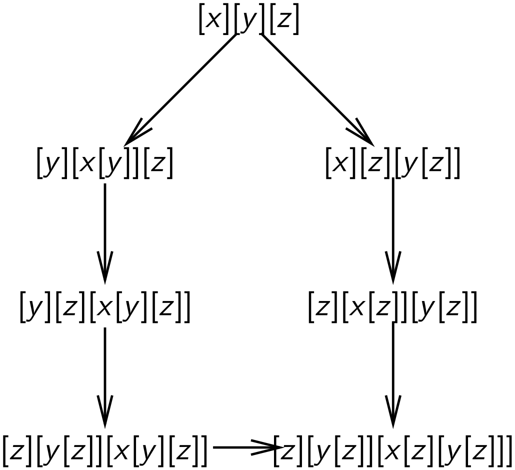
Interpret as substitution lemma

$$M[x{:=}N][y{:=}P] \to M[y{:=}P][x{:=}N[y{:=}P]]$$

$xyz \to xz(yz)$ critical pair



$[y][z] \to [z][y[z]]$ critical pair



Conclusion

- Constructive confluence (tiling)

Conclusion

- Constructive confluence (tiling)
- via local confluence (decreasing diagrams)

Conclusion

- Constructive confluence (tiling)
- via local confluence (decreasing diagrams)
- via orthogonality (residual systems)

Conclusion

- Constructive confluence (tiling)
- via local confluence (decreasing diagrams)
- via orthogonality (residual systems)
- complexity ?