The FMC from a higher-order rewriting perspective

a syntactician’s prology

Vincent van Oostrom[1]

https://people.bath.ac.uk/vvo21/

[1] Supported by EPSRC Project EP/R029121/1 Typed lambda-calculi with sharing and unsharing.

A rewriter's perspective

Dialogue

- Q_1: can you say something about my calculus?

A rewriter's perspective

Dialogue

- Q_1: can you say something about my calculus?
- Q_2: what are the objects A and what are the rules P?

A rewriter's perspective

Dialogue

- Q_1: can you say something about my calculus?
- Q_2: what are the objects A and what are the rules P?
- A_2: terms over $\{A, 0, S\}$, rules $A(x, 0) \to x$ and $A(x, S(y)) \to S(A(x, y))$

A rewriter’s perspective

Dialogue

- Q_1: can you say something about my calculus?
- Q_2: what are the objects A and what are the rules P?
- A_2: terms over $\{A, 0, S\}$, rules $A(x, 0) \to x$ and $A(x, S(y)) \to S(A(x, y))$
- A_1: what would you like to know about it?

Birkhoff theorem as Leitmotiv

$$
\begin{tikzcd}
 & P \models t = s \arrow[dl, "\text{sound \& complete}"'] \arrow[dr] & \\
P \vdash t = s \arrow[rr, dash, "\text{logicality}"'] & & t \leftrightarrow^*_P s
\end{tikzcd}
$$

interest in?

equational theory (refl),(sym),(trans),(compatible),(rule) ? (Birkhoff)

Sub-Birkhoff theorem as Leitmotiv

$$
\begin{tikzcd}
 & P \models t \geq s \arrow[dl, "\text{sound \& complete}"'] \arrow[dr] & \\
P \vdash t \geq s \arrow[rr, dash, "\text{logicality}"'] & & t \twoheadrightarrow_P s
\end{tikzcd}
$$

interest in?

rewrite theory (refl),(trans),(compatible),(rule) ? (Meseguer)

Sub-Birkhoff theorem as Leitmotiv

$$
\begin{tikzcd}
& P \models t > s \arrow[dl, "\text{sound \& complete}"'] \arrow[dr] & \\
P \vdash t > s \arrow[rr, dash, "\text{logicality}"'] & & t \to_P^+ s
\end{tikzcd}
$$

interest in?

termination theory (trans),(compatible),(rule),well-founded? (Lankford,Zantema)

Sub-Birkhoff theorem as Leitmotiv

$$\begin{tikzcd}
& P \models t > s \arrow[dl, "\text{sound \& complete}"'] \arrow[dr] & \\
P \vdash t > s \arrow[rr, dash, "\text{logicality}"'] & & t \to_P^+ s
\end{tikzcd}$$

interest in?

any other sub-equational theory $\subseteq$ $(\mathit{refl}), (\mathit{sym}), (\mathit{trans}), (\mathit{compatible}), (\mathit{rule})$?

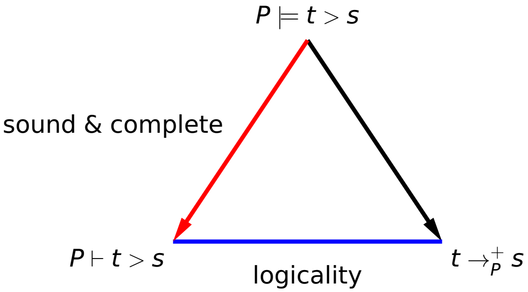
Sub-Birkhoff theorem as Leitmotiv

$$\begin{tikzcd} & P \models t > s \arrow[dl, "\text{sound \& complete}"'] \arrow[dr] & \\ P \vdash t > s \arrow[rr, dash, "\text{logicality}"'] & & t \to_P^+ s \end{tikzcd}$$

interest in?

computations? represent as terms; standardisation $\Rightarrow$ (2D; Klop,Melliès)

Sub-Birkhoff theorem as Leitmotiv



interest in?

approximation? infinitary terms / rewriting (Klop,Ariola,Blom,Ketema)

Higher-order rewrite system (HRS) warm-up examples

the relevance of arbitrary signatures

combinatory logic (CL)	:	term rewrite system (TRS)
. .		. .
lambda-calculus (lambda)	:	higher-order term rewrite system (HRS; Nipkow)

closed under renaming, adding recursion / algebraic rules, etc.

freeness: signature $\Longrightarrow$ terms, signature $+$ rules $\Longrightarrow$ steps

- terms: simply typed λ-terms modulo $\alpha\beta\eta$ over typed signature
- steps: simply typed λ-terms modulo $\alpha\beta\eta$ over typed signature & typed rules source (target) by mapping each rule $\rho : \ell \to r$ in step to lhs ℓ (rhs r)

Higher-order rewrite system warm-up examples

Example (addition as a HRS)

- signature $0 : o$ (nullary), $S : o \to o$ (unary), $A : o \to o \to o$ (binary)

Higher-order rewrite system warm-up examples

Example (addition as a HRS)

- signature $0 : o$ (nullary), $S : o \to o$ (unary), $A : o \to o \to o$ (binary)
- rules $\rho : o \to o$ and $\theta : o \to o \to o$, for variables $x, y : o$:

$$\begin{array}{lll} \rho\text{:} \quad \lambda x.A\,x\,0 & \to & \lambda x.x \\ \theta\text{:} \lambda xy.A\,x\,(S\,y) & \to & \lambda xy.S\,(A\,x\,y) \end{array}$$

cf. Frege's shift from $\forall x.t = s$ to $\lambda x.t = \lambda x.s$

Higher-order rewrite system warm-up examples

Example (addition as a HRS)

- signature $0 : o$ (nullary), $S : o \to o$ (unary), $A : o \to o \to o$ (binary)
- rules $\rho : o \to o$ and $\theta : o \to o \to o$, for variables $x, y : o$

$$\begin{array}{lrcl} \rho: & \lambda x.A(x,0) & \to & \lambda x.x \\ \theta: & \lambda xy.A(x,S(y)) & \to & \lambda xy.S(A(x,y)) \end{array}$$

with syntactic sugar added

Higher-order rewrite system warm-up examples

Example (untyped lambda-beta-eta calculus as a HRS)

- signature abs : $(o \to o) \to o$ (higher-order), app : $o \to o \to o$

Higher-order rewrite system warm-up examples

Example (untyped lambda-beta-eta calculus as a HRS)

- signature abs : $(o \to o) \to o$ (higher-order), app : $o \to o \to o$
- rules eta : $o \to o$, beta : $(o \to o) \to o \to o$, variables $M : o \to o$ and $N, K : o$

$$\begin{array}{rrcl} \text{eta:} & \lambda K.\text{abs}\,\lambda x.\text{app}\,K\,x & \to & \lambda K.K \\ \text{beta:} & \lambda MN.\text{app}\,(\text{abs}\,\lambda x.M\,x)\,N & \to & \lambda MN.M\,N \end{array}$$

without syntactic sugar; x is parameter to M; K no parameters

Higher-order rewrite system warm-up examples

Example (untyped lambda-beta-eta calculus as a HRS)

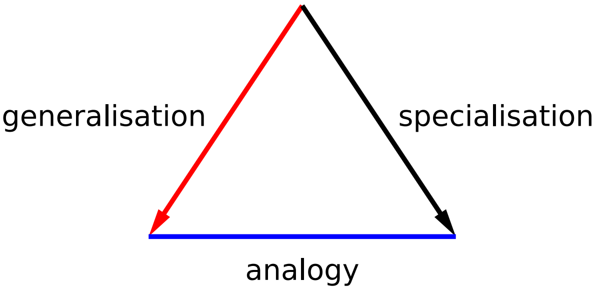
- signature abs : $(o \to o) \to o$ (higher-order), app : $o \to o \to o$
- rules eta : $o \to o$, beta : $(o \to o) \to o \to o$, variables $M : o \to o$ and $N, K : o$

$$\begin{array}{rrcl} \text{eta:} & \lambda K.\text{abs}(\lambda x.\text{app}(K, x)) & \to & \lambda K.K \\ \text{beta:} & \lambda MN.\text{app}(\text{abs}(\lambda x.M(x)), N) & \to & \lambda MN.M(N) \end{array}$$

with syntactic sugar

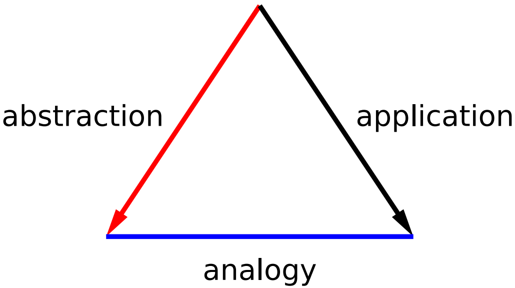
Steps as terms over signature + rules

Pólya’s triangle



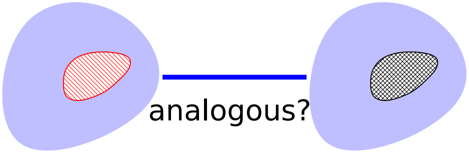
Steps as terms over signature + rules

Pólya’s triangle

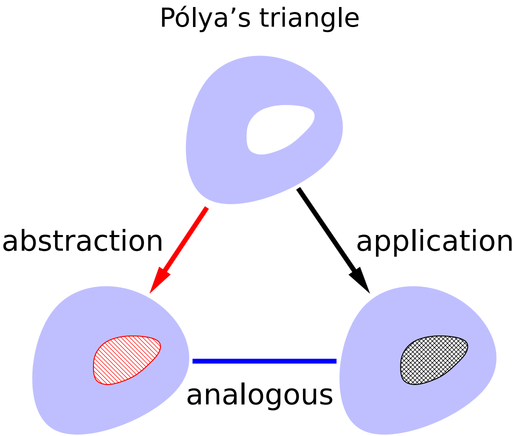


Steps as terms over signature + rules

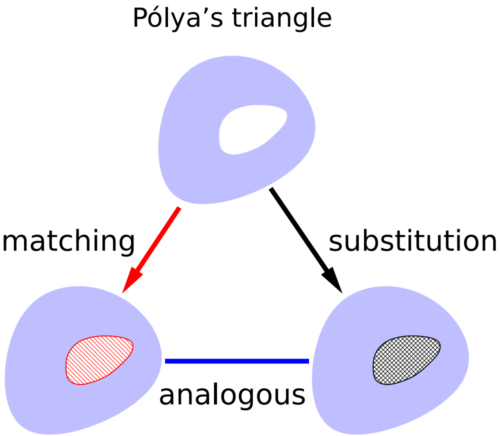
Pólya's triangle



Steps as terms over signature + rules

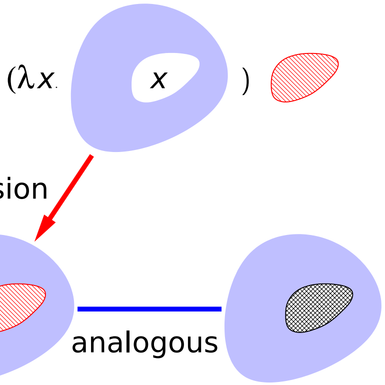


Steps as terms over signature + rules



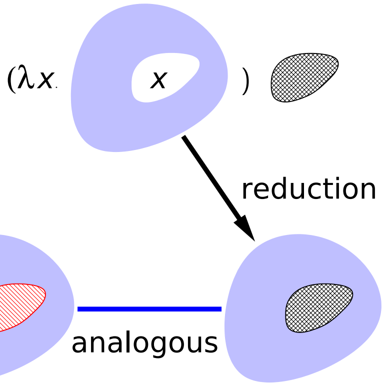
Steps as terms over signature + rules

structured rewrite step for rule $\rho : \ell \to r$



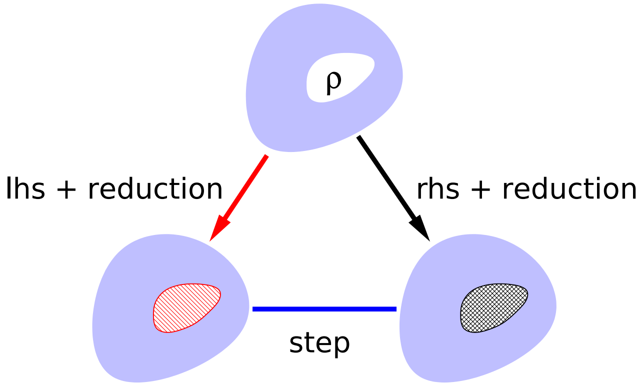
Steps as terms over signature + rules

structured rewrite step for rule ρ : $\ell \to r$



Steps as terms over signature + rules

structured rewrite step for rule ρ : $\ell \to r$



Steps as terms over signature + rules

Example (Steps in HRS for addition)

$$\begin{array}{rcl} \rho\text{:} \quad \lambda x.A(x,0) & \to & \lambda x.x \\ \theta\text{:}\lambda xy.A(x,S(y)) & \to & \lambda xy.S(A(x,y)) \end{array}$$

- $S(\rho(0))$ step from $S((\lambda x.A(x,0))\,0){\downarrow} = S(A(0,0))$ to $S((\lambda x.x)\,0){\downarrow} = S(0)$

Steps as terms over signature + rules

Example (Steps in HRS for addition)

$$\begin{aligned} \rho\colon \quad \lambda x.A(x,0) &\to \lambda x.x \\ \theta\colon \lambda xy.A(x,S(y)) &\to \lambda xy.S(A(x,y)) \end{aligned}$$

- $S(\rho(0))$ step from $S((\lambda x.A(x,0))\,0){\downarrow} = S(A(0,0))$ to $S((\lambda x.x)\,0){\downarrow} = S(0)$
- $\rho(\theta(0,0))$ multistep from $(\lambda x.A(x,0))\,((\lambda xy.A(x,S(y)))\,0\,0){\downarrow} = A(A(0,S(0)),0)$ to $(\lambda x.x)((\lambda xy.S(A(x,y)))\,0\,0){\downarrow} = S(A(0,0))$

Steps as terms over signature + rules

Example (Steps in HRS for addition)

$$\rho\!: \quad \lambda x.A(x,0) \;\to\; \lambda x.x$$
$$\theta\!:\lambda xy.A(x,S(y)) \;\to\; \lambda xy.S(A(x,y))$$

- $S(\rho(0))$ step from $S((\lambda x.A(x,0))\,0)\!\downarrow\, = S(A(0,0))$ to $S((\lambda x.x)\,0)\!\downarrow\, = S(0)$
- $\rho(\theta(0,0))$ multistep from $(\lambda x.A(x,0))\,((\lambda xy.A(x,S(y)))\,0\,0)\!\downarrow\, = A(A(0,S(0)),0)$ to $(\lambda x.x)((\lambda xy.S(A(x,y)))\,0\,0)\!\downarrow\, = S(A(0,0))$

Remark (steps as terms non-standard still)

simply typed λ-calculus modulo $\alpha\beta\eta$ for binding / matching / substitution
adaptable to strings, graphs, . . . ; can also be replaced by proof nets, . . .

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$\begin{aligned} \lambda x.A\,x\,0 &\to_\rho \lambda x.x \\ \lambda xy.A\,x\,(S\,y) &\to_\theta \lambda xy.S\,(A\,x\,y) \end{aligned}$$

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$\begin{array}{rcl} \lambda x.A\,x\,0 & \to_\rho & \lambda x.x \\ \lambda xy.A\,x\,(S\,y) & \to_\theta & \lambda xy.S\,(A\,x\,y) \end{array}$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$

- interpret base type o as $\mathbb{N}$ ($[\![o]\!] := \mathbb{N}$), $\tau \to \sigma$ as set $[\![\tau]\!] \Rightarrow [\![\sigma]\!]$ of functions from $[\![\tau]\!]$ to $[\![\sigma]\!]$, function application / abstraction according to their name

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$\lambda x.A\,x\,0 \quad \to_\rho \quad \lambda x.x$$
$$\lambda xy.A\,x\,(S\,y) \quad \to_\theta \quad \lambda xy.S\,(A\,x\,y)$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$

- interpret each symbol $f : \tau$ as an element of its type $[\![\tau]\!]$, say 0 as $37 \in \mathbb{N}$, S as id $\in \mathbb{N} \Rightarrow \mathbb{N}$, A as first projection $\pi_1 \in \mathbb{N} \Rightarrow \mathbb{N} \Rightarrow \mathbb{N}$

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$
\begin{aligned}
\lambda x.A\,x\,0 &\to_\rho \lambda x.x \\
\lambda xy.A\,x\,(S\,y) &\to_\theta \lambda xy.S\,(A\,x\,y)
\end{aligned}
$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$

- interpret each symbol $f : \tau$ as an element of its type $[\![\tau]\!]$, say 0 as $37 \in \mathbb{N}$, S as id $\in \mathbb{N} \Rightarrow \mathbb{N}$, A as first projection $\pi_1 \in \mathbb{N} \Rightarrow \mathbb{N} \Rightarrow \mathbb{N}$
- interpret rules ρ and θ as equalities

$$
\begin{aligned}
(n \mapsto n) &= (n \mapsto n) \\
(n, m \mapsto n) &= (n, m \mapsto n)
\end{aligned}
$$

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$
\begin{aligned}
\lambda x.A\,x\,0 &\to_\rho \lambda x.x \\
\lambda xy.A\,x\,(S\,y) &\to_\theta \lambda xy.S\,(A\,x\,y)
\end{aligned}
$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$

- interpret each symbol $f : \tau$ as an element of its type $[\![\tau]\!]$, say 0 as $37 \in \mathbb{N}$, S as id $\in \mathbb{N} \Rightarrow \mathbb{N}$, A as first projection $\pi_1 \in \mathbb{N} \Rightarrow \mathbb{N} \Rightarrow \mathbb{N}$
- interpret rules ρ and θ as equalities

$$
\begin{aligned}
(n \mapsto n) &= (n \mapsto n) \\
(n, m \mapsto n) &= (n, m \mapsto n)
\end{aligned}
$$

of course interpreting as zero, successor, and addition also works

Semantics of / via higher-order term rewriting?

semantics of addition HRS?

$$\begin{aligned} \lambda x.A\,x\,0 &\to_\rho \lambda x.x \\ \lambda xy.A\,x\,(S\,y) &\to_\theta \lambda xy.S\,(A\,x\,y) \end{aligned}$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$

- interpret each symbol $f : \tau$ as an element of its type $[\![\tau]\!]$, say 0 as $37 \in \mathbb{N}$, S as id $\in \mathbb{N} \Rightarrow \mathbb{N}$, A as first projection $\pi_1 \in \mathbb{N} \Rightarrow \mathbb{N} \Rightarrow \mathbb{N}$
- interpret rules ρ and θ as equalities

$$\begin{aligned} (n \mapsto n) &= (n \mapsto n) \\ (n, m \mapsto n) &= (n, m \mapsto n) \end{aligned}$$

two, successor, and multiplication gives inequalities $>$ on $\mathbb{N}_{\geq 2}$ (termination)

Semantics of / via higher-order term rewriting?

semantics of untyped lambda-beta-eta HRS?

$$\lambda\,(K).\mathsf{abs}\,\lambda x.\mathsf{app}\,K\,x \quad \rightarrow_{\mathsf{eta}} \quad \lambda\,(K).K$$

$$\lambda\,(M\,N).\mathsf{app}\,(\mathsf{abs}\,\lambda x.M\,x)\,N \quad \rightarrow_{\mathsf{beta}} \quad \lambda\,(M\,N).M\,N$$

Semantics of / via higher-order term rewriting?

semantics of untyped lambda-beta-eta HRS?

$$\begin{array}{rcl} \lambda\,(K).\mathsf{abs}\,\lambda x.\mathsf{app}\,K\,x & \to_{\mathsf{eta}} & \lambda\,(K).K \\ \lambda\,(M\,N).\mathsf{app}\,(\mathsf{abs}\,\lambda x.M\,x)\,N & \to_{\mathsf{beta}} & \lambda\,(M\,N).M\,N \end{array}$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$; CCC

- interpret beta and eta-rules in CCC (cf. Koymans):

$$\begin{array}{rcl} @ \circ \langle [\![\mathsf{abs}]\!] \circ \langle\,\rangle, @ \circ \langle [\![\mathsf{app}]\!] \circ \langle\,\rangle, \mathsf{id}\rangle\rangle & = & \mathsf{id} \\ @ \circ \langle [\![\mathsf{app}]\!] \circ \langle\,\rangle, @ \circ \langle [\![\mathsf{abs}]\!] \circ \langle\,\rangle, \mathsf{id}\rangle\rangle & = & \mathsf{id} \end{array}$$

Semantics of / via higher-order term rewriting?

semantics of untyped lambda-beta-eta HRS?

$$
\begin{array}{rcl}
\lambda(K).\mathsf{abs}\,\lambda x.\mathsf{app}\,K\,x & \to_{\mathsf{eta}} & \lambda(K).K \\
\lambda(M\,N).\mathsf{app}\,(\mathsf{abs}\,\lambda x.M\,x)\,N & \to_{\mathsf{beta}} & \lambda(M\,N).M\,N
\end{array}
$$

factorise through semantics of substitution; simply typed $\lambda\beta\eta$; CCC

- interpret beta and eta-rules in CCC (cf. Koymans):

$$
\begin{array}{rcl}
@ \circ \langle [\![\mathsf{abs}]\!] \circ \langle\,\rangle, @ \circ \langle [\![\mathsf{app}]\!] \circ \langle\,\rangle, \mathsf{id} \rangle\rangle & = & \mathsf{id} \\
@ \circ \langle [\![\mathsf{app}]\!] \circ \langle\,\rangle, @ \circ \langle [\![\mathsf{abs}]\!] \circ \langle\,\rangle, \mathsf{id} \rangle\rangle & = & \mathsf{id}
\end{array}
$$

- for set / functions: $[\![\mathsf{abs}]\!] \circ [\![\mathsf{app}]\!] = \mathsf{id}$ on D and $[\![\mathsf{app}]\!] \circ [\![\mathsf{abs}]\!] = \mathsf{id}$ on $D \Rightarrow D$

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \;\longrightarrow\; H.\,\{N/x\}M$$

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \;\rightharpoonup\; H.\,\{N/x\}M$$

Defects of standard presentation from a HRS perspective

- terms given by grammar, with external notion of binding

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\, M \mid [N]a.\, M \mid a\langle x\rangle.\, M$$

- rule

$$[N]a.\, H.\, a\langle x\rangle.\, M \;\rightarrow\; H.\, \{N/x\}M$$

Defects of standard presentation from a HRS perspective

- terms given by grammar, with external notion of binding
- rule schema: H sequences of abs/apps at $b \neq a$ and $\{N/x\}M$ meta-level

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \;\rightharpoonup\; H.\,\{N/x\}M$$

Defects of standard presentation from a HRS perspective

- terms given by grammar, with external notion of binding
- rule schema: H sequences of abs/apps at $b \neq a$ and $\{N/x\}M$ meta-level
- steps allow rule applications in any context; how defined exactly?

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \rightharpoonup H.\,\{N/x\}M$$

Desiderata to embed in higher-order term rewrite system

- simply typed $\lambda\beta\eta\alpha$-terms freely generated from typed signature; λ-binding
- rule schema: H sequences of abs/apps at $b \neq a$ and $\{N/x\}M$ meta-level
- steps allow rule applications in any context; how defined exactly?

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \;\rightarrow\; H.\,\{N/x\}M$$

Desiderata to embed in higher-order term rewrite system

- simply typed $\lambda\beta\eta\alpha$-terms freely generated from typed signature; λ-binding
- typed, closed rules; variables and substitutions at object-level
- steps allow rule applications in any context; how defined exactly?

FMC in a standard presentation

Definition (the FMC)

- terms

$$M, N, P \quad ::= \quad \star \mid x.\,M \mid [N]a.\,M \mid a\langle x\rangle.\,M$$

- rule

$$[N]a.\,H.\,a\langle x\rangle.\,M \;\rightarrow\; H.\,\{N/x\}M$$

Desiderata to embed in higher-order term rewrite system

- simply typed $\lambda\beta\eta\alpha$-terms freely generated from typed signature; λ-binding
- typed, closed rules; variables and substitutions at object-level
- steps freely generated from signature extended with rules

FMC in a standard presentation: substitution

Definition (substitution $\{M/x\}N$)

is capture-avoiding, uses composition $N;M$ (also capture avoiding):

$$
\begin{array}{rclrcll}
\star\,;M & := & M & [P].\,N\,;M & := & [P].\,(N\,;M) & \\
x.\,N\,;M & := & x.\,(N\,;M) & \langle y\rangle.\,N\,;M & := & \langle y\rangle.\,(N\,;M) & (y\notin \mathrm{fv}(M))
\end{array}
$$

$$
\begin{array}{rcll rcll}
\{P/x\}\star & := & \star & & \{P/x\}[N]a.\,M & := & [\{P/x\}N]a.\,\{P/x\}M & \\
\{P/x\}x.\,M & := & P;\{P/x\}M & & \{P/x\}a\langle x\rangle.\,M & := & a\langle x\rangle.\,M & \\
\{P/x\}y.\,M & := & y.\,\{P/x\}M & (x\neq y) & \{P/x\}a\langle y\rangle.\,M & := & a\langle y\rangle.\,\{P/x\}M & (y\notin \mathrm{fv}(P))
\end{array}
$$

FMC in a standard presentation: substitution

Definition (substitution $\{M/x\}N$)

is capture-avoiding, uses composition $N;M$ (also capture avoiding):

$$\begin{aligned} \star\,;M &:= M & [P].N\,;M &:= [P].(N\,;M) \\ x.N\,;M &:= x.(N\,;M) & \langle y\rangle.N\,;M &:= \langle y\rangle.(N\,;M) \quad (y \notin \mathrm{fv}(M)) \end{aligned}$$

$$\begin{aligned} \{P/x\}\star &:= \star & \{P/x\}[N]a.M &:= [\{P/x\}N]a.\{P/x\}M \\ \{P/x\}x.M &:= P;\{P/x\}M & \{P/x\}a\langle x\rangle.M &:= a\langle x\rangle.M \\ \{P/x\}y.M &:= y.\{P/x\}M \quad (x \neq y) & \{P/x\}a\langle y\rangle.M &:= a\langle y\rangle.\{P/x\}M \quad (y \notin \mathrm{fv}(P)) \end{aligned}$$

how to deal with composition in HRS?

mark tip of P by bound variable $\chi \implies$ composition is substitution for χ

FMC as a (third-order) HRS

Definition ($\mathcal{FMC}$)

- signature: $\mathsf{lam}_a : ((o \to o) \to o) \to o$ and $\mathsf{app}_a : o \to (o \to o) \to o$ for every a

FMC as a HRS

Definition ($\mathcal{FMC}$)

- signature: $\mathsf{lam}_a : ((o \to o) \to o) \to o$ and $\mathsf{app}_a : o \to (o \to o) \to o$ for every a
- rules: for variables $\vec{K}$ free in H, and $\vec{x}$ bound there, $x, \vec{x} : o \to o$

  $$\mathsf{beta}_{H,a} : \lambda \vec{K} M N.\mathsf{app}_a(H[\mathsf{lam}_a(\lambda x.M(\vec{x},x))], \lambda \chi.N(\chi)) \to \lambda \vec{K} M N.H[M(\vec{x}, \lambda \chi.N(\chi))]$$

FMC as a HRS

Definition ($\mathcal{FMC}$)

- signature: $\mathsf{lam}_a : ((o \to o) \to o) \to o$ and $\mathsf{app}_a : o \to (o \to o) \to o$ for every a
- rules: for variables $\vec{K}$ free in H, and $\vec{x}$ bound there, $x, \vec{x} : o \to o$

  $\mathsf{beta}_{H,a} : \lambda \vec{K}MN.\mathsf{app}_a(H[\mathsf{lam}_a(\lambda x.M(\vec{x}, x))], \lambda \chi.N(\chi)) \to \lambda \vec{K}MN.H[M(\vec{x}, \lambda \chi.N(\chi))]$

Lemma (FMC embedding $\langle\,\rangle$)

in fragment $\lambda \chi.S$ *with* $S ::= \chi \mid x\,S \mid \mathsf{app}_a(S, \lambda \chi.S) \mid \mathsf{lam}_a(\lambda x.S)$

- $\star$ *maps to* χ
- $x.\,M$ *maps to* $x\langle M \rangle$
- $[N]a.\,M$ *maps to* $\mathsf{app}_a(\langle M \rangle, \lambda \chi.\langle N \rangle)$

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof
2. is FMC computation $\twoheadrightarrow_\beta$ a (partial) function?
   yes, confluence by Okui's multi–one critical pair criterion

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof
2. is FMC computation $\twoheadrightarrow_\beta$ a (partial) function?
   yes, confluence by Okui's multi–one critical pair criterion
3. is equational theory $=_{\text{beta}}$ consistent (non-trivial model)?
   yes, because Church–Rosser and distinct normal forms (Church–Rosser)

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof
2. is FMC computation $\twoheadrightarrow_\beta$ a (partial) function?
   yes, confluence by Okui's multi–one critical pair criterion
3. is equational theory $=_{\text{beta}}$ consistent (non-trivial model)?
   yes, because Church–Rosser and distinct normal forms (Church–Rosser)
4. do we have good strategies?
   yes, spine reduction is hyper-normalising by random descent

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof
2. is FMC computation $\twoheadrightarrow_\beta$ a (partial) function?
   yes, confluence by Okui's multi–one critical pair criterion
3. is equational theory $=_{\text{beta}}$ consistent (non-trivial model)?
   yes, because Church–Rosser and distinct normal forms (Church–Rosser)
4. do we have good strategies?
   yes, spine reduction is hyper-normalising by random descent
5. is the combination with eta well-behaved?
   yes, commutes with beta by critical pair criterion

Properties of FMC via HRS theory for $\mathcal{FMC}$?

Potentially interesting questions

1. is $>_\beta$ well-founded (termination model)?
   yes, for typed FMC by Gandy-proof
2. is FMC computation $\twoheadrightarrow_\beta$ a (partial) function?
   yes, confluence by Okui's multi–one critical pair criterion
3. is equational theory $=_{\text{beta}}$ consistent (non-trivial model)?
   yes, because Church–Rosser and distinct normal forms (Church–Rosser)
4. do we have good strategies?
   yes, spine reduction is hyper-normalising by random descent
5. is the combination with eta well-behaved?
   yes, commutes with beta by critical pair criterion
6. reductions modulo permutation equivalence a computation category?
   yes, because multisteps $\multimap\!\!\rightarrow_{\text{beta}}$ constitute residual system (CTS; Stark)

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\rightarrow_{\text{beta}}$ *is confluent*

Definition

- rewrite system $\rightarrow := \langle A, \Phi, \text{src}, \text{tgt} \rangle$
  $\phi : \ a \rightarrow b$ or $a \rightarrow_\phi b$ denotes step ϕ with source $\text{src}(\phi) = a$, target $\text{tgt}(\phi) = b$
  (rewrite systems have same data as multigraphs, quivers, pre-categories)

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\to_{\text{beta}}$ *is confluent*

Definition

- rewrite system $\to := \langle A, \Phi, \text{src}, \text{tgt} \rangle$
- has diamond property if $\forall$ peak $b \leftarrow a \to c$, $\exists$ valley $b \to d \leftarrow c$
  Skolemisation: $\forall \phi, \psi\ \text{src}(\phi) = \text{src}(\psi) \implies \text{tgt}(\psi/\phi) = \text{tgt}(\phi/\psi)$ (residuation)

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\rightarrow_{\text{beta}}$ *is confluent*

Definition

- rewrite system $\rightarrow \; := \langle A, \Phi, \text{src}, \text{tgt} \rangle$
- has diamond property if $\forall$ peak $b \leftarrow a \rightarrow c$, $\exists$ valley $b \rightarrow d \leftarrow c$
- is confluent if reflexive–transitive closure $\twoheadrightarrow$ has diamond

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\rightarrow_{\text{beta}}$ *is confluent*

Difference between beta **in $\mathcal{FMC}$ and lambda-calculus**

- beta rule in lambda-calculus is orthogonal; all occurrences concurrent
- beta rule in $\mathcal{FMC}$ is non-orthogonal; (schematic) self-overlaps:

$$\mathsf{app}_a\text{–}\mathsf{app}_b\text{–}\mathsf{lam}_b\text{–}\mathsf{lam}_a$$

$$\mathsf{app}_b\text{–}\mathsf{app}_a\text{–}\mathsf{lam}_b\text{–}\mathsf{lam}_a$$

  these are harmless; idea: beta does not change H; then use:

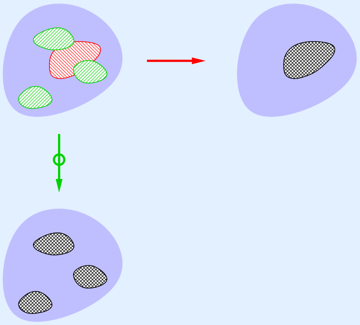
- $\rightarrow$ confluent if $\rightarrow\;\subseteq\;\multimap\!\!\rightarrow\;\subseteq\;\twoheadrightarrow$ and $\forall$ peaks $b \leftarrow\!\!\circ\!\!- a \rightarrow c$, $\exists$ valley $b \twoheadrightarrow d \leftarrow\!\!\circ\!\!- c$

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\to$ *is confluent if* $\forall$ *critical* $b \leftarrow\!\!\circ\!\!- a \to c$, $\exists\, b \twoheadrightarrow d \leftarrow\!\!\circ\!\!- c$

Proof by potatoes of Okui's criterion (for $-\!\!\circ\!\!\to_{\text{beta}}$ and $\to_{\text{beta}}$).



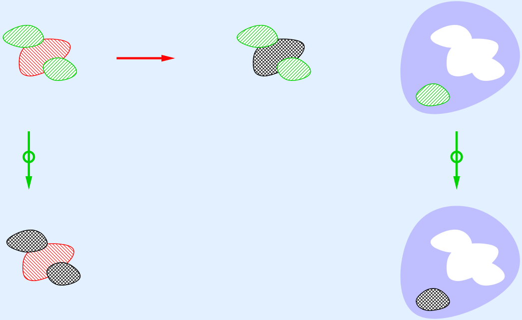
$\square$

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\to$ *is confluent if* $\forall$ *critical* $b \leftarrow\!\!\circ\!\!- \, a \to c$, $\exists\, b \twoheadrightarrow d \leftarrow\!\!\circ\!\!- \, c$

Proof by potatoes of Okui's criterion (for $\multimap_{\mathrm{beta}}$ and $\to_{\mathrm{beta}}$).



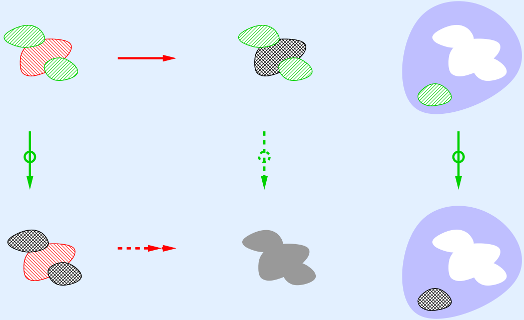
$\square$

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\rightarrow$ *is confluent if* $\forall$ *critical* $b \leftarrow\!\circ\!- \ a \rightarrow c$, $\exists\, b \twoheadrightarrow d \leftarrow\!\circ\!- \ c$

Proof by potatoes of Okui's criterion (for $-\!\circ\!\!\rightarrow_{\text{beta}}$ and $\rightarrow_{\text{beta}}$).



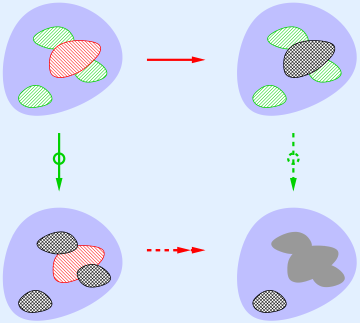
$\square$

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\to$ *is confluent if* $\forall$ *critical* $b \leftarrow\!\!\circ\!\!- a \to c$, $\exists\, b \twoheadrightarrow d \leftarrow\!\!\circ\!\!- c$

Proof by potatoes of Okui's criterion (for $-\!\!\circ\!\!\to_{\text{beta}}$ and $\to_{\text{beta}}$).



$\square$

Confluence of $\mathcal{FMC}$ by Okui's critical pair criterion

Theorem

$\to$ *is confluent if* $\forall$ *critical* $b \leftarrow\!\!\circ\!\!- a \to c$, $\exists\, b \twoheadrightarrow d \leftarrow\!\!\circ\!\!- c$

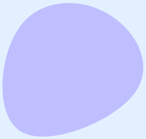
Main challenge : formalise this

- any overlapping multi–one peak $t \leftarrow\!\!\circ\!\!- s \to r$
- decomposes as $(\lambda x.D)\,\hat{t} \leftarrow\!\!\circ\!\!- (\lambda x.C)\,\hat{s} \to (\lambda x.C)\,\hat{r}$ for multi–one critical peak $\hat{t} \leftarrow\!\!\circ\!\!- \hat{s} \to \hat{r}$ and multistep $D \leftarrow\!\!\circ\!\!- C$
- for multi–one critical peak $\hat{t} \leftarrow\!\!\circ\!\!- \hat{s} \to \hat{r}$ exists many–multi valley $\hat{t} \twoheadrightarrow \hat{u} \leftarrow\!\!\circ\!\!- \hat{r}$
- recomposing with multistep $D \leftarrow\!\!\circ\!\!- C$ yields many–multi valley $(\lambda x.D)\,\hat{t} \to (\lambda x.D)\,\hat{u} \leftarrow\!\!\circ\!\!- (\lambda x.C)\,\hat{r}$

Geometric vs. inductive pattern

Idea: allow to carve out well-behaved part, pat $\iff$ pattern

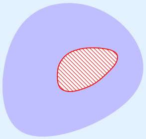
given a term



Geometric vs. inductive pattern

Idea: allow to carve out well-behaved part, pat $\Longleftrightarrow$ pattern

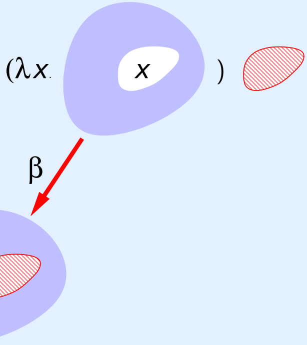
select convex set of edges and nodes, a pat P (geometric)



Geometric vs. inductive pattern

Idea: allow to carve out well-behaved part, pat $\Longleftrightarrow$ pattern

β-expand to occurrence of pattern π (inductive)



Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

- if an @ is in P so is left child, and that is not a λ-abstraction (passive params)

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

- if an @ is in P so is left child, and that is not a λ-abstraction
- taking n left childs of @s from root yields head symbol $f : \tau^n \to o$

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

- if an @ is in P so is left child, and that is not a λ-abstraction
- taking n left childs of @s from root yields head symbol $f : \tau^n \to o$
- if variable position is in P then also its binder position (closed)

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

- closed simply typed λ-term

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

- closed simply typed λ-term
- in long-β-normal form (type can be read-off from term)

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

- closed simply typed λ-term
- in long-β-normal form
- has function symbol as head

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

- closed simply typed λ-term
- in long-β-normal form
- has function symbol as head
- if shape $\lambda \vec{F}.s$, then parameters $\vec{F}$ occur in that order in t

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

- closed simply typed λ-term
- in long-β-normal form
- has function symbol as head
- if shape $\lambda\vec{F}.s$, then parameters $\vec{F}$ occur in that order in t
- each F_i in it has variables ($\neq \vec{F}$) bound above as arguments in that order

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

Theorem

bijection between pats and pattern-occurrences in a term

Geometric vs. inductive pattern

Definition (pat; geometric)

non-empty convex set P of node and edge positions in tree of a λ-term

Definition (pattern occurrence; inductive)

$(\lambda x.t)\,\pi$ occurrence of π in $(\lambda x.t)\,\pi{\downarrow_\beta}$ if x once in t with π:

Theorem

bijection between pats and pattern-occurrences in a term

Example

lhs $\lambda FS.\mathrm{app}(\mathrm{abs}(\lambda x.F(x)), S)$ of rule beta of $\mathcal{FMC}$ is a pattern

Geometric vs. inductive patterns

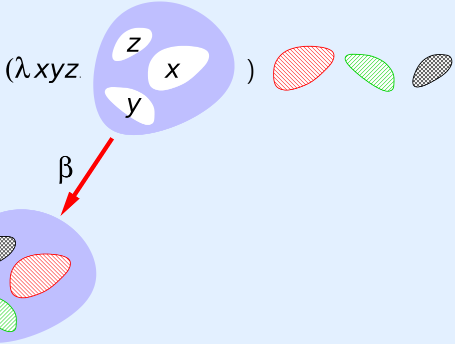
Theorem (distributive lattice)

- *sets of pats wrt subset (of union)*
- *occurrences of vectors of patterns wrt refinement*

Geometric vs. inductive patterns

Idea

bijection extends to family of non-overlapping pats

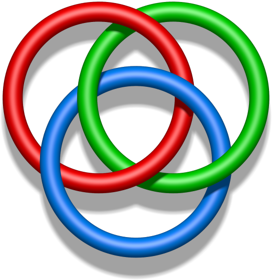


Geometric vs. inductive patterns

Theorem (distributive lattice)

- *sets of pats wrt subset (of union)*
- *occurrences of vectors of patterns wrt refinement*

in particular no Borromean rings situation



Conclusions / questions

- we have said something (FMC meta-theory via HRS results for $\mathcal{FMC}$)

Conclusions / questions

- we have said something (FMC meta-theory via HRS results for $\mathcal{FMC}$)
- can we say more?

Conclusions / questions

- we have said something (FMC meta-theory via HRS results for $\mathcal{FMC}$)
- can we say more?
- FMC semantics via $\mathcal{FMC}$? surely coding of stacks too coarse; linear types?

Conclusions / questions

- we have said something (FMC meta-theory via HRS results for $\mathcal{FMC}$)
- can we say more?
- FMC semantics via $\mathcal{FMC}$? surely coding of stacks too coarse; linear types?
- rule instead of rule schema? rule pattern is regular language

Conclusions / questions

- we have said something (FMC meta-theory via HRS results for $\mathcal{FMC}$)
- can we say more?
- FMC semantics via $\mathcal{FMC}$? surely coding of stacks too coarse; linear types?
- rule instead of rule schema? rule pattern is regular language
- work modulo permutation to make beta, eta local? (almost no HRS modulo)