# Reversible Computation and Principal Types in $\lambda^!$-calculus

F.Alessi, A.Ciaffaglione, P.Di Gianantonio, F.Honsell,
M.Lenisa, I.Scagnetto
Department of Mathematics, Computer Science, and Physics
University of Udine

**Logic Colloquium 2018** Udine, 27$^{th}$ July 2018

**Logic Colloquium '82** in Florence
Furio Honsell and Marco Forti
*A Model where Cardinal Ordering is Universal*

# Introduction I

- Ordinary computation is irreversible, it dissipates energy.
- Yet, in principle, apart from IO-operations, it is *per se* reversible.
- In the '90's, J.Y.Girard invented Geometry of Interaction (GoI) for modeling Linear Logic and $\lambda$-calculus.
- Abramsky showed how GoI gives rise to game semantics for logics and hence also universal models of computation consisting of strategies, not necessarily winning, in extremely stylized games.

# Introduction II

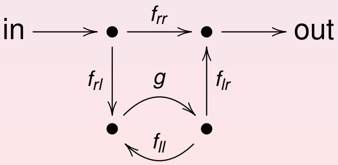
- Abramsky showed also how to model $\lambda$-calculus using partial involutions over suitable languages of moves *i.e.* reversible functions $f(i) = j \leftrightarrow f(j) = i$.
- Involutions amount to history-free strategies and apply according to Abramsky's Joyal's GoI symmetric feedback, or equivalently, Girard's Execution Formula.
- Since the behaviour on moves of the strategies interpreting, say 0 and 1, easily discriminates them, we can check reversibly the behaviour of computable characteristic functions.
- Thus game models of partial involutions can be viewed as a reversible universal model of computation.

# Abramky's Model of Reversible Computation

An affine combinatory algebra arising from a GoI situation:

Partial Involutions, $\mathcal{P}$

- $T_\Sigma$, the language of moves: $\Sigma_0 = \{\epsilon\}$, $\Sigma_1 = \{l, r\}$, $\Sigma_2 = \{<\ ,\ >\}$; terms $r(x)$ are input words, while terms $l(x)$ are output words;
- partial involutions over $T_\Sigma$: partial injective functions $f : T_\Sigma \rightharpoonup T_\Sigma$ such that $f(u) = v \Leftrightarrow f(v) = u$;
- replication: $!f = \{(< t, u >, < t, v >) \mid t \in T_\Sigma \wedge (u, v) \in f\}$;
- linear application: $f \cdot g = f_{rr} \bigcup (f_{rl}; g; (f_{ll}; g)^*; f_{lr})$, where $f_{ij} = \{(u, v) | (i(u), j(v)) \in f\}$ for $i, j \in \{r, l\}$.



# Our starting point

- The model of partial involutions is a special case of a general categorical paradigm [Abamsky-Haghverdi-Scott02, Abramsky-Lenisa05, Haghverdi00], called Abramsky's Programme, which amounts to defining a *combinatory algebra* and then a $\lambda$-algebra starting from a Geometry of Interaction (GoI) Situation in a traced symmetric monoidal category.

$$\text{GoI situation} \longrightarrow \text{combinatory algebra} \overset{\text{quotient}}{\longrightarrow} \lambda\text{-algebra}$$

- From a GoI situation to a combinatory algebra is a well explored construction, many examples *e.g.* square summable sequences $l_2$, partial functions, partial injective functions.
- But the last step, from a combinatory algebra to a $\lambda$-algebra and the relation to the model theory of $\lambda$-calculus had not been fully analyzed yet.

# Some Questions

We recall that a $\lambda$-algebra is a categorical model of $\lambda$-calculus, *i.e.* a reflexive object in a Cartesion Closed Category.

- Can all GoI combinatory algebras be quotiented to $\lambda$-algebras?
- Is the combinatory algebra of partial involutions a $\lambda$-algebra or can it be quotiented to one?
- Abramsky's Question [Abramsky05]: characterise the partial involutions that are denotations of combinators.

# Outline of our work

- In our paper Linear $\lambda$-calculus and Reversible Automatic Combinators (available at https://arxiv.org/abs/1806.06759) we investigate Abramsky's algebras from the point of view of the model theory of $\lambda$-calculus.
- We focus on the strictly linear and strictly affine parts of Abramky's algebra of partial involutions, *i.e.* without replication.
- We provide full answers to the questions above, in terms of strictly linear/affine combinatory logic, $\lambda$-calculus, and corresponding BCI/BCK-combinatory and BCI/BCK-$\lambda$-algebras.
- **NO**, **YES**, for strictly linear, **NO** already for the strictly linear case, but **YES** all such algebras can be quotiented to one.
- BUT in the process of doing this is discovered **MUCH MORE**

# Main Results

- We highlight a **duality** between the **GoI interpretation** of a $\lambda$-term as a partial involution and its **principal type** w.r.t. an intersection types discipline for a linear/affine $\lambda$-calculus.
- We unveil three conceptually independent, but ultimately equivalent, accounts of application in the $\lambda$-calculus:
  - $\beta$**-reduction**,
  - GoI application of involutions based on Joyal's symmetric feedback / Girard's Execution Formula,
  - **unification** of principal types.
- In order to prove that the model of partial involutions is a BCI-$\lambda$-algebra, we use an implementation in Erlang of the application of involutions.

# First Step: Linear/Affine Combinatory Logic and Combinatory Algebras

Linear/Affine Combinatory Logic $\mathbf{CL^L}$ ($\mathbf{CL^A}$):

- variables $x, y, \ldots$,
- $\dfrac{M \in \mathbf{CL^X} \quad N \in \mathbf{CL^X}}{MN \in \mathbf{CL^X}}$ for $X \in \{L, A\}$
- combinators $B, C, I$ (and $K$ in the affine case) satisfying

$$BMNP = M(NP) \qquad IM = M \qquad CMNP = (MP)N \qquad KMN = M$$

Linear/Affine Combinatory Algebras (BCI-algebra, BCK-algebra):
$\mathcal{A} = (A, \cdot)$ with combinators $B, C, I$ (and $K$ in the affine case) satisfying the equations.

Interpretation of $\mathbf{CL}^X$ into a combinatory algebra $\mathcal{A}$:
$[\![\ ]\!]_{\mathcal{A}} : \mathbf{CL}^X \longrightarrow \mathcal{A}$

# Linear/Affine Lambda Calculus

$\Lambda^L$: $$\frac{M \in \Lambda^L \quad N \in \Lambda^L}{MN \in \Lambda^L} \qquad \frac{M \in \Lambda^L \quad \mathcal{E}(x, M)}{\lambda x.M \in \Lambda^L}$$

$\Lambda^A$: $$\frac{M \in \Lambda^A \quad N \in \Lambda^A}{MN \in \Lambda^A} \qquad \frac{M \in \Lambda^A \quad \mathcal{O}(x, M)}{\lambda x.M \in \Lambda^A}$$

$\mathcal{E}(x, M)$: $x$ appears free in $M$ exactly once
$\mathcal{O}(x, M)$: $x$ appears free in $M$ at most once.

$(\beta_L)$ $(\lambda x.M)N \to M[N/x]$ $\qquad (\xi_L)$ $$\frac{M = N \quad \mathcal{E}(x, M) \quad \mathcal{E}(x, N)}{\lambda x.M = \lambda x.N}$$

$(\beta_A)$ $(\lambda x.M)N \to M[N/x]$ $\qquad (\xi_A)$ $$\frac{M = N \quad \mathcal{O}(x, M) \quad \mathcal{O}(x, N)}{\lambda x.M = \lambda x.N}$$

# From $\mathbf{CL^A}$ to $\Lambda^A$ and back

$( )_{\lambda^A} : \mathbf{CL^A} \to \Lambda^A$

$$(B)_{\lambda^A} = \lambda xyz.x(yz) \qquad (I)_{\lambda^A} = \lambda x.x$$
$$(C)_{\lambda^A} = \lambda xyz.(xz)y \qquad (K)_{\lambda^A} = \lambda xy.x$$

$( )_{CL^A} : \Lambda^A \to \mathbf{CL^A}$ replaces each $\lambda$-abstraction by a $\lambda^*$-abstraction.

Abstraction Operation:

$\lambda^* x.x = I \quad \lambda^* x.c = Kc \quad \lambda^* x.y = Ky$

$$\lambda^* x.MN = \begin{cases} C(\lambda^* x.M)N & \text{if } x \in FV(M) \\ BM(\lambda^* x.N) & \text{if } x \in FV(N)\,. \\ K(MN) & \text{otherwise}\,. \end{cases}$$

**Theorem (Affine Abstraction)**

$(\lambda^* x.M)N = M[N/x]$.

# Affine $\lambda$-algebra

- Let $\mathcal{A} = (A, \cdot)$ be an affine combinatory algebra (ACA). Define $\mathcal{T}(\mathcal{A})$ as the set of terms of $\mathrm{CL}^A$ extended with constants $c_a$, for $a \in A$.
- An ACA $\mathcal{A}$ is an affine $\lambda$-algebra if $\forall M, N \in \mathcal{T}(\mathcal{A})$,

$$\vdash M_\lambda =_\lambda N_\lambda \implies [\![M]\!]_{\mathcal{A}} = [\![N]\!]_{\mathcal{A}} \ ,$$

- Equivalently, a purely equational definition à la Curry: equations $A^\beta$ ensuring that

  if $\mathbf{CL^A} + A^\beta \vdash M = N$ then $\vdash [\![\lambda^* x.M]\!]_{\mathcal{A}} = [\![\lambda^* x.N]\!]_{\mathcal{A}}$.
  (Closure under the $\xi$-rule as a rule of proof)

# Affine $\lambda$-algebra: equational definition

**Theorem**

*An affine $\lambda$-algebra $\mathcal{A}$ is an ACA satisfying the following sets of equations:*

- $$B = \lambda^* xyz.x(yz) = \lambda^* xyz.Bxyz$$ $$C = \lambda^* xyz.(xz)y = \lambda^* xyz.Cxyz$$ $$I = \lambda^* x.x = \lambda^* x.Ix$$ $$K = \lambda^* xy.x = \lambda^* xy.Kxy$$
- *equations for* $\lambda^* x.IP = \lambda^* x.P$ *to hold:* $\lambda^* y.BIy = \lambda^* yz.yz$
- *equations for* $\lambda^* x.BPQR = \lambda^* x.P(QR)$ *to hold:*
  - $\lambda^* uvw.C(C(BBu)v)w = \lambda^* uvw.Cu(vw)$
  - $\lambda^* uvw.C(B(Bu)v)w = \lambda^* uvw.Bu(Cvw)$
  - $\lambda^* uvw.B(Buv)w = \lambda^* uvw.Bu(Bvw)$
- *for* $\lambda^* x.CPQR = \lambda^* x.PRQ$ *to hold: ...*
- *for* $\lambda^* x.KPQ = \lambda^* x.P$ *to hold: ...*
- *2 more equations are necessary for K in dealing with* $\xi$ *over axioms: ...*

# From Combinatory Algebras to $\lambda$-algebras

### Proposition

*(i) Not all ACAs are affine $\lambda$-algebras.*
*(ii) Let $\mathcal{A} = (A, \cdot)$ be an ACA. The quotient $(A/\equiv_{\mathcal{A}}, \cdot_{\equiv_{\mathcal{A}}})$ is an affine $\lambda$-algebra, where*
*$a \equiv_{\mathcal{A}} b$ iff $\exists$ closed $M, N \in \mathcal{T}(\mathcal{A})$ s.t. $a = [\![M]\!]_{\mathcal{A}}$, $b = [\![N]\!]_{\mathcal{A}}$, and $(M)_{\lambda^A} =_{\lambda^A} (N)_{\lambda^A}$.*
*(iii) Not all non-trivial ACA's can be quotiented to a non-trivial affine $\lambda$-algebra.*

### Proof.

(i) The closed term model of affine CL, *e.g.* CKK $\neq$ I. The algebra of partial involutions $\mathcal{P}$.
(ii) $\equiv_{\mathcal{A}}$ is a congruence w.r.t. application.
(iii) *E.g.* the closed term model of the ACA induced by the equations: $(SII)(SII) = I \quad (S(BII)(BII))(S(BII)(BII)) = K$. □

# Simple Types for Linear $\lambda$-calculus

**Definition (Simple Types)**

$(Type \ni)\ \mu ::= \alpha \mid \mu \to \mu$

**Definition (Linear Type Discipline)**

$$\frac{}{x : \mu \vdash_L x : \mu} \qquad \frac{x \in FV(M) \quad \Gamma, x : \mu \vdash_L M : \nu}{\Gamma \vdash_L \lambda x.M : \mu \to \nu}$$

$$\frac{\Gamma \vdash_L M : \mu \to \nu \quad \Delta \vdash_L N : \mu \quad (dom(\Gamma) \cap dom(\Delta)) = \emptyset}{\Gamma, \Delta \vdash_L MN : \nu}$$

# Principal Types

## Definition (Principal Type Scheme)

$$\frac{}{x : \alpha \Vdash_L x : \alpha} \qquad \frac{x \in FV(M) \quad \Gamma, x : \mu \Vdash_L M : \nu}{\Gamma \Vdash_L \lambda x.M : \mu \to \nu}$$

$$\frac{\begin{array}{c}\Gamma \Vdash_L M : \mu \quad \Delta \Vdash_L N : \tau \quad (dom(\Gamma) \cap dom(\Delta)) = \emptyset \\ (TVar(\Gamma) \cap TVar(\Delta)) = \emptyset \quad (TVar(\mu) \cap TVar(\tau)) = \emptyset \\ U' = MGU(\mu, \alpha \to \beta) \quad U = MGU(U'(\alpha), \tau) \quad \alpha, \beta \text{ fresh}\end{array}}{U(\Gamma, \Delta) \Vdash_L MN : U \circ U'(\beta)}$$

## Definition ($MGU(\sigma, \tau)$)

$$\frac{MGU(\alpha, \tau) = U \quad \alpha \in TVar \quad \tau \notin TVar}{MGU(\tau, \alpha) = U} \qquad \frac{\alpha \in TVar \quad \alpha \notin \tau}{MGU(\alpha, \tau) = id[\tau/\alpha]}$$

$$\frac{MGU(\sigma_1, \tau_1) = U_1 \quad MGU(U_1(\sigma_2), U_1(\tau_2)) = U_2}{MGU(\sigma_1 \to \sigma_2, \tau_1 \to \tau_2) = U_2 \circ U_1}$$

# Properties of Linear Types

## Theorem

*(i)* $\Gamma \Vdash_L M : \sigma \Longrightarrow$ *each type variable occurs at most twice in* $\Gamma \Vdash_L M : \sigma$.
*(ii)* $\Gamma \Vdash_L M : \sigma \Longrightarrow \forall\, U$ *s.t. each type variable occurs at most twice in* $U(\Gamma), U(\sigma)$, $U(\Gamma) \vdash_L M : U(\sigma)$;
*(iii)* $\Gamma \vdash_L M : \sigma \Longrightarrow \exists\, \Gamma' \Vdash_L M : \sigma'$ *and a type substitution* $U$ *s.t.* $U(\Gamma') = \Gamma$ *and* $U(\sigma') = \sigma$.

Examples of principal types:

| | | |
|---|---|---|
| $I$ | $\lambda x.x$ | $\alpha \to \alpha$ |
| $B$ | $\lambda xyz.x(yz)$ | $(\alpha \to \gamma) \to (\beta \to \alpha) \to \beta \to \gamma$ |
| $C$ | $\lambda xyz.xzy$ | $(\alpha \to \beta \to \gamma) \to \beta \to \alpha \to \gamma$ |

## Theorem (Linear Subject Conversion)

$M =_{\beta^L} M'$ *and* $\Gamma \Vdash_L M : \sigma \Longrightarrow \Gamma \Vdash_L M' : \sigma$.

# The Affine Case

Extend $\vdash$ with: $$\frac{\Gamma \vdash_A M : \nu \quad x \notin dom(\Gamma) \quad TVar(\mu) \text{ fresh}}{\Gamma \vdash_A \lambda x.M : \mu \to \nu}$$

Extend $\Vdash$ with: $$\frac{\Gamma \Vdash_A M : \nu \quad x \notin dom(\Gamma) \quad \alpha \text{ fresh}}{\Gamma \Vdash_A \lambda x.M : \alpha \to \nu}$$

But subject conversion fails:

- $\not\Vdash_A \lambda xyz.(\lambda w.x)(yz) : \alpha_1 \to \alpha_2 \to \alpha_3 \to \alpha_1$
- but only $\Vdash_A \lambda xyz.(\lambda w.x)(yz) : \alpha_1 \to (\alpha_2 \to \alpha_3) \to \alpha_2 \to \alpha_1$, which is an instance of the former.
- But $\vdash \lambda xyz.x : \alpha_1 \to \alpha_2 \to \alpha_3 \to \alpha_1$.

This is the root reason why the affine algebra of partial involutions is not a $\lambda$-algebra.

- Combinators:

$$
\begin{array}{lll}
B & : & r^3x \leftrightarrow lrx\ ,\ l^2x \leftrightarrow rlrx\ ,\ rl^2x \leftrightarrow r^2lx \\
C & : & l^2x \leftrightarrow r^2lx\ ,\ lrlx \leftrightarrow rlx\ ,\ lr^2x \leftrightarrow r^3x \\
I & : & lx \leftrightarrow rx \\
K & : & lx \leftrightarrow r^2x
\end{array}
$$

- A concrete device for inducing combinators: pattern-matching bi-orthogonal automata.
- Reversibility: the behaviour of a functional program can be rendered by applying reversibly the corresponding automaton to suitable inputs.

# From Principal Types to Involutions (and back)

## Principal Types

| | | |
|---|---|---|
| $I$ | $\lambda x.x$ | $\alpha \to \alpha$ |
| $K$ | $\lambda xy.x$ | $\alpha \to (\beta \to \alpha)$ |
| $B$ | $\lambda xyz.x(yz)$ | $(\alpha \to \gamma) \to ((\beta \to \alpha) \to (\beta \to \gamma))$ |
| $C$ | $\lambda xy.xzy$ | $(\alpha \to (\beta \to \gamma)) \to (\beta \to (\alpha \to \gamma))$ |

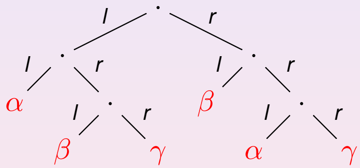
## Partial Involutions

$$
\begin{array}{lcl}
I & : & lx \leftrightarrow rx \\
K & : & lx \leftrightarrow r^2x \\
B & : & r^3x \leftrightarrow lrx \ , \ l^2x \leftrightarrow rlrx \ , \ rl^2x \leftrightarrow r^2lx \\
C & : & l^2x \leftrightarrow r^2lx \ , \ lrlx \leftrightarrow rlx \ , \ lr^2x \leftrightarrow r^3x
\end{array}
$$

# Example

## Principal Type

$C \quad \lambda xy.xzy \quad (\alpha \to (\beta \to \gamma)) \to (\beta \to (\alpha \to \gamma))$




## Partial Involution

$C \quad : \quad l^2x \leftrightarrow r^2lx \ , \ lrlx \leftrightarrow rlx \ , \ lr^2x \leftrightarrow r^3x$

# From Principal Types to Involutions: the algorithm

Let $\Vdash_A M : \mu$.
For $\alpha \in \mu$, the judgements $\mathcal{T}(\alpha, \mu)$ yield a pair in the graph of a partial involution, if $\alpha$ occurs twice in $\mu$, or an element of $T_\Sigma$, if $\alpha$ occurs once in $\mu$:

$$\begin{array}{rcl} \mathcal{T}(\alpha,\alpha) & = & \alpha \\ \mathcal{T}(\alpha, \mu(\alpha) \to \nu(\alpha)) & = & l(\mathcal{T}(\alpha, \mu(\alpha))) \leftrightarrow r(\mathcal{T}(\alpha, \nu(\alpha))) \\ \mathcal{T}(\alpha, \mu(\alpha) \to \nu) & = & l[\mathcal{T}(\alpha, \mu(\alpha))] \\ \mathcal{T}(\alpha, \mu \to \nu(\alpha)) & = & r[\mathcal{T}(\alpha, \nu(\alpha))] \end{array}$$

where $r[x] = \begin{cases} rx_1 \leftrightarrow rx_2 & \text{if } x = x_1 \leftrightarrow x_2 \ \wedge \ x_1, x_2 \in T_\Sigma \\ rx & \text{otherwise} \end{cases}$
and similarly for $l[x]$.

Partial involution: $f_\mu = \{\mathcal{T}(\alpha, \mu) \mid \alpha \text{ appears twice in } \mu\}$.

Vice versa, any partial involution interpreting a closed $\mathbf{CL^A}$-term induces the corresponding principal type.

## Theorem

*(i) The linear combinatory algebra of partial involutions is a linear $\lambda$-algebra, albeit not a linear combinatory $\lambda$-model.*
*(ii) The affine combinatory algebra of partial involutions is not an affine $\lambda$-algebra.*

Proof

(i) Using the Erlang implementation.
(ii)

- $(BBK)_{\lambda^A} = (BKK)_{\lambda^A}$,
- but $[\![BBK]\!]_{\mathcal{P}} \neq [\![BKK]\!]_{\mathcal{P}}$.
- Namely,
  $[\![BBK]\!]_{\mathcal{P}} = [\![\lambda^* xyz.Kx(yz)]\!]_{\mathcal{P}} = \{lx \leftrightarrow r^3x, rl^2x \leftrightarrow r^2lx\}$
- while $[\![BKK]\!]_{\mathcal{P}} = [\![\lambda^* xyz.x]\!]_{\mathcal{P}} = \{lx \leftrightarrow r^3x\}$.

# Abramsky's Question

Question [Abramsky05]: Characterise partial involutions which are denotations of combinators.

Answer: In the affine case, the denotations of combinators are the partial involutions from which we can synthesise a principal type scheme which is a tautology in minimal logic.

**Proposition**

*If $\mu$ type s.t.*

- *each type variable occurs at most twice*
- *inhabited by M closed*

*then*
$\exists N.\ \Gamma \Vdash_L N : \mu$ *and* $N =_{\beta_L \eta} M$.

Examples:

- $H_1 = \{lllx \leftrightarrow rllx, llrx \leftrightarrow lrx, rrx \leftrightarrow rlx\}$ partial involution
  $((\alpha \to \beta) \to \gamma) \to (\alpha \to \beta) \to \gamma$ type
  $\lambda xy.x(\lambda z.yz)$ $\eta$-expansion of the identity.

- $H_2 = \{lllx \leftrightarrow lrrx, llrx \leftrightarrow lrlx, lrrx \leftrightarrow rrrx\}$ partial involution
  $((\alpha \to \beta) \to (\beta \to \alpha)) \to \gamma \to \gamma$ type
  $\lambda yx.(\lambda w.x)(\lambda zw.yzw)$ $\beta$-expansion of $\lambda xy.y$.

# Examples: partial involutions not denoting $\lambda$-terms

- $H_3 = \{lx \leftrightarrow rlrx, r^2x \leftrightarrow rl^2x\}$ partial involution
  $\beta \to (\alpha \to \beta) \to \alpha$ type
  partial involutions are reversible, $\lambda$-terms are not.
- $H_4 = \{r^3x \leftrightarrow l^2x, lrx \leftrightarrow rlx\}$ partial involution
  $(\alpha \to \beta) \to (\beta \to \gamma \to \alpha)$ type
  $\lambda x^{\alpha\to\beta}.\lambda y^{\beta}.\lambda z^{\gamma}.?$.

# Second Step: The Extension to the Full Linear case I- Affine Combinatory Logic

Affine Combinatory Logic $\mathbf{CL}^!$ includes variables $x, y, \ldots$, distinguished constants (combinators) $B, C, I, K, W, D, \delta, F$, and it is closed under application and promotion, *i.e.*:

$$\frac{M \in \mathbf{CL}^! \quad N \in \mathbf{CL}^!}{MN \in \mathbf{CL}^!} \qquad \frac{M \in \mathbf{CL}^!}{!M \in \mathbf{CL}^!}$$

Combinators satisfy the following equations:

$$\begin{array}{llll} BMNP = M(NP) & IM = M & CMNP = (MP)N & KMN = M \\ WM!N = M!N!N & \delta!M = !!M & D!M = M & F!M!N = !(MN) \end{array}$$

Affine Combinatory algebra

$$\begin{array}{llll} Bxyz = x(yz) & Ix = x & Cxyz = (xz)y & Kxy = x \\ Wx!y = x!y!y & \delta!x = !!x & D!x = x & F!x!y = !(xy). \end{array}$$

# Second Step: The Extension to the Full Linear case - $\lambda^!$-calculus

Affine $\lambda$!-calculus:

$$\frac{M \in \Lambda^! \quad N \in \Lambda^!}{MN \in \Lambda^!} \qquad \frac{M \in \Lambda^!}{!M \in \Lambda^!} \qquad \frac{M \in \Lambda^! \quad \mathcal{O}_!(x, M)}{\lambda x.M \in \Lambda^!}$$

$$\frac{M \in \Lambda^!}{\lambda !x.M \in \Lambda^!}\ ,$$

where $\mathcal{O}_!(x, M)$ means that the $x$ appears in $M$ at most once, and not in the scope of a !.

The rules are the restrictions of the standard $\beta$-rule and $\xi$-rule to linear abstractions, the pattern-$\beta$-reduction rule, the $str$!-structural rule, and the $\xi$!-rule, namely:

$$(\beta_r) \ \ (\lambda x.M)N \to M[N/x] \qquad (\beta !) \ \ (\lambda !x.M)!N \to M[N/x]$$

$$(\xi_r) \quad \frac{M = N \quad \lambda x.M, \lambda x.N \in \Lambda^!}{\lambda x.M = \lambda x.N} \qquad (str!) \quad \frac{M = N}{!M = !N}$$

$$(\xi !) \quad \frac{M = N}{\lambda !x.M = \lambda !x.N}\ .$$

# Second Step: The Extension to the Full Linear case - Abstraction

The Abstraction Operation is defined by induction on $M \in \mathbf{CL}^!$:
$\lambda^*!x.x = D \quad \lambda^*!x.c = Kc \quad \lambda^*!x.y = Ky \quad \lambda^*!x.!x = F(!I)$, for $M \not\equiv x$
$\lambda^*!x.MN = W(B(C\lambda^*!x.M)(\lambda^*!x.N))$
$\lambda^*!x.!M = B(F(!\lambda^*!x.M))\delta$
$\lambda^*x.x = I \quad \lambda^*x.c = Kc \quad \lambda^*x.y = Ky$

$$\lambda^*x.MN = \begin{cases} C(\lambda^*x.M)N & \text{if } x \in FV(M), \\ BM(\lambda^*x.N) & \text{if } x \in FV(N), \\ K(MN) & \text{otherwise.} \end{cases}$$

# Second Step: The Extension to the Full Linear case - !Intersection Type Discipline

The type discipline for the affine $\lambda$!-calculus is defined as follows:

**Definition (!Intersection Types)**

$(Type \ni)\ \mu ::= \alpha \mid \mu \to \nu \mid !_u\mu \mid \mu \wedge \nu$
where $\alpha$ denotes a type variable in $TVar$, and $u \in T_\Sigma[X]$.

# Second Step: The Extension to the Full Linear case - !Intersection Types Examples

| | | |
|---|---|---|
| $I$ | $\lambda x.x$ | $\alpha \to \alpha$ |
| $K$ | $\lambda xy.x$ | $\alpha \to \beta \to \alpha$ |
| $B$ | $\lambda xyz.x(yz)$ | $(\alpha \to \gamma) \to (\beta \to \alpha) \to \beta \to \gamma$ |
| $C$ | $\lambda xy.xzy$ | $(\alpha \to \beta \to \gamma) \to \beta \to \alpha \to \gamma$ |
| $D$ | $\lambda !x.x$ | $!_{\epsilon}\alpha \to \alpha$ |
| $\delta$ | $\lambda !x.!!x$ | $!_{<u,v>}\alpha \to !_u!_v\alpha$ |
| $F$ | $\lambda !x!y.!(xy)$ | $!_u(\alpha \to \beta) \to !_u\alpha \to !_u\beta$ |
| $W$ | $\lambda x!_u y.x!_u y!_u y$ | $(!\alpha \to !\beta \to \gamma) \to (!\alpha \wedge !\beta) \to \gamma$ |
| $\Delta$ | $\lambda !x.x!x$ | $!_u(!_v\alpha \to \beta) \wedge !_v\alpha \to \beta$ |
| $2$ | $\lambda !f.\lambda x.f(fx))$ | $!_u(\alpha \to \beta) \wedge !_u(\beta \to \gamma) \to \alpha \to \gamma$ |

# Second Step: The Extension to the Full Linear case - Partial Involutions corresponding to the Examples

$$
\begin{array}{lcl}
B & : & r^3x \leftrightarrow lrx \ , \ l^2x \leftrightarrow rlrx \ , \ rl^2x \leftrightarrow r^2lx \\
C & : & l^2x \leftrightarrow r^2lx \ , \ lrlx \leftrightarrow rlx \ , \ lr^2x \leftrightarrow r^3x \\
F & : & l\langle x, ry\rangle \leftrightarrow r^2\langle x, y\rangle \ , \ l\langle x, ly\rangle \leftrightarrow rl\langle x, y\rangle \\
W & : & r^2x \leftrightarrow lr^2x \ , \ l^2\langle x, y\rangle \leftrightarrow rl\langle lx, y\rangle \ , \ lrl\langle x, y\rangle \leftrightarrow rl\langle rx, y\rangle \\
I & : & lx \leftrightarrow rx \\
K & : & lx \leftrightarrow r^2x \\
\delta & : & l\langle\langle x, y\rangle, z\rangle \leftrightarrow r\langle x, \langle y, z\rangle\rangle \\
D & : & l\langle e, x\rangle \leftrightarrow rx \, .
\end{array}
$$

# Current Work

- Proof that the quotient on the affine combinatory algebra is indeed a $\lambda$-algebra.
- Characterise the theory induced by the model of partial involutions.