

Propositions as Sessions

Philip Wadler

University of Edinburgh

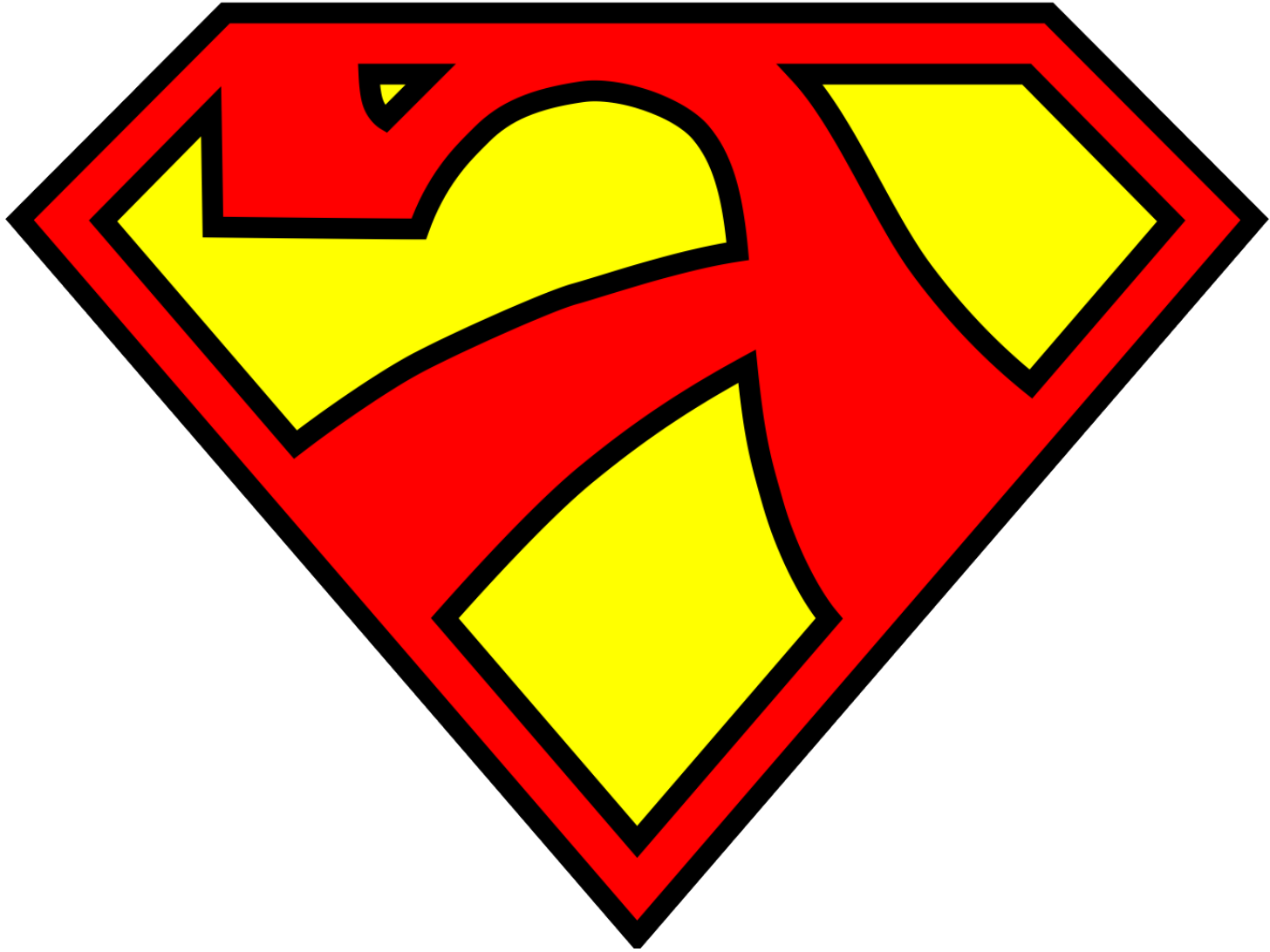
Summer School on Trends in Computing

Tarragona, Friday 10 July 2014

Kohei Honda, 1959–2012



EPSRC Programme Grant EP/K034413/1
From Data Types to Session Types:
A Basis for Concurrency and Distribution (ABCD)
Simon Gay, Nobuko Yoshida, Philip Wadler



Propositions as Types

propositions	<i>as</i>	types
proofs	<i>as</i>	programs
normalisation of proofs	<i>as</i>	evaluation of programs

Propositions as Types is robust

propositions	<i>as</i>	types
proofs	<i>as</i>	programs
normalisation of proofs	<i>as</i>	evaluation of programs
Intuitionistic Natural Deduction	\leftrightarrow	Simply-Typed Lambda Calculus
Quantification over propositions	\leftrightarrow	Polymorphism
Quantification over individuals	\leftrightarrow	Dependent types
Modal Logic	\leftrightarrow	Monads (state, exceptions)
Classical-Intuitionistic Embedding	\leftrightarrow	Continuation Passing Style

... but there's a missing link

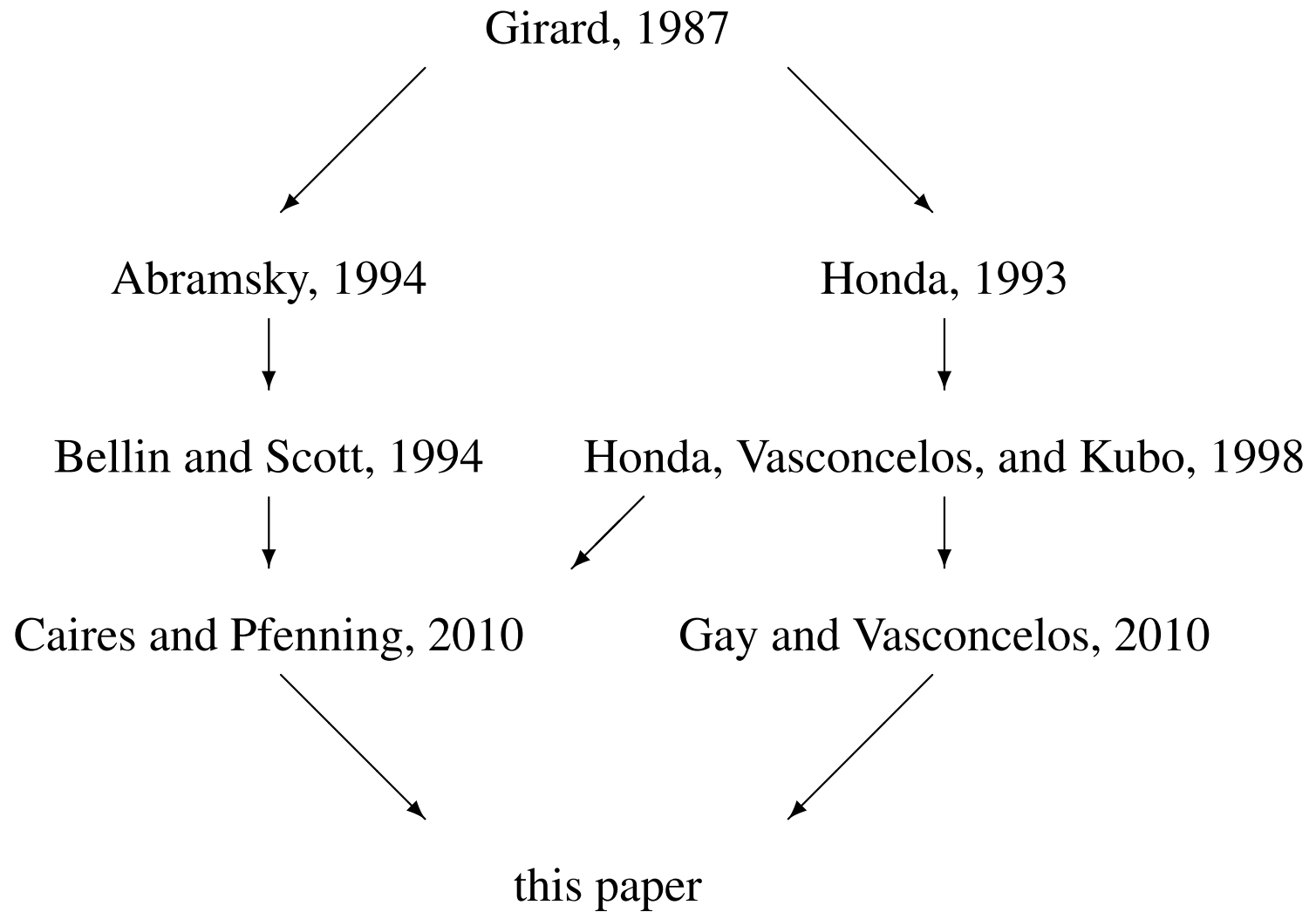
propositions	<i>as</i>	types
proofs	<i>as</i>	programs
normalisation of proofs	<i>as</i>	evaluation of programs
Intuitionistic Natural Deduction	\leftrightarrow	Simply-Typed Lambda Calculus
Quantification over propositions	\leftrightarrow	Polymorphism
Quantification over individuals	\leftrightarrow	Dependent types
Modal Logic	\leftrightarrow	Monads (state, exceptions)
Classical-Intuitionistic Embedding	\leftrightarrow	Continuation Passing Style
???	\leftrightarrow	Process Calculus

Propositions as Sessions

propositions	<i>as</i>	types
proofs	<i>as</i>	programs
normalisation of proofs	<i>as</i>	evaluation of programs

propositions	<i>as</i>	session types
proofs	<i>as</i>	processes
cut elimination	<i>as</i>	communication

Lines of development



The Twist

- Abramsky, 1994: Proofs as Processes

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, z : B}{\nu y, z. x\langle y, z \rangle. (P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

$$\frac{R \vdash \Theta, y : A, z : B}{x(y, z). R \vdash \Theta, x : A \wp B} \wp$$

- this paper: Propositions as Sessions

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{\nu y. x\langle y \rangle. (P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

$$\frac{R \vdash \Theta, y : A, x : B}{x(y). R \vdash \Theta, x : A \wp B} \wp$$

A small change in notation

- With $\nu x. x\langle y \rangle$

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{\nu y. x\langle y \rangle. (P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

- With $x[y]$

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{x[y]. (P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

ILL vs. CLL

- Caires and Pfenning, 2010: Intuitionistic Linear Logic

$$\frac{\Gamma; \Delta \vdash P :: y : A \quad \Gamma; \Delta' \vdash Q :: x : B}{\Gamma; \Delta, \Delta' \vdash \nu y. x \langle y \rangle. (P \mid Q) :: x : A \otimes B} \otimes\text{-R}$$

$$\frac{\Gamma; \Delta \vdash P :: y : A \quad \Gamma; \Delta', x : B \vdash Q :: z : C}{\Gamma; \Delta, \Delta', x : A \multimap B \vdash \nu y. x \langle y \rangle. (P \mid Q) :: z : C} \multimap\text{-L}$$

$$\frac{\Gamma; \Delta, y : A \vdash R :: x : B}{\Gamma; \Delta \vdash x(y).R :: x : A \multimap B} \multimap\text{-R}$$

$$\frac{\Gamma; \Delta, y : A, x : B \vdash R :: z : C}{\Gamma; \Delta, x : A \otimes B \vdash x(y).R :: z : C} \otimes\text{-L}$$

- this paper: Classical Linear Logic

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{x[y].(P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

$$\frac{R \vdash \Theta, y : A, x : B}{x(y).R \vdash \Theta, x : A \wp B} \wp$$

Axiom and Polymorphism

- Abramsky, 1994

$$\frac{}{x(z).w\langle z\rangle.0 \vdash w : X^\perp, x : X} \text{Ax}$$

- Caires and Pfenning, 2010

(no axiom)

- this paper (based on an idea from Caires and Pfenning, 2011)

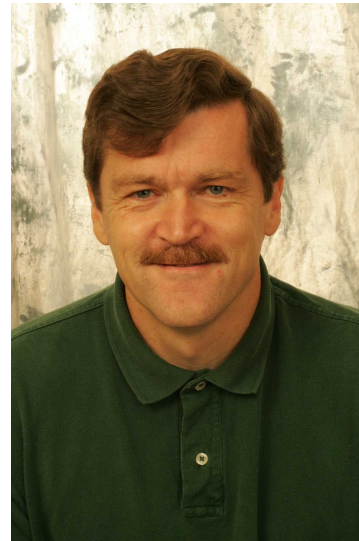
$$\frac{}{w \leftrightarrow x \vdash w : A^\perp, x : A} \text{Ax}$$

Part I

CP

Classical Processes

Caires-Pfenning



Cut Elimination

Theorem

(Subject Reduction)

If $P \vdash \Gamma$ and $P \Longrightarrow Q$ then $Q \vdash \Gamma$.

(Cut Elimination)

If $P \vdash \Gamma$ then there exists a Q
such that $P \Longrightarrow^* Q$ and Q is not a Cut.

Types

$A, B, C ::=$

X type variable

$A \otimes B$ output A then behave as B

$A \oplus B$ select from A or B

$!A$ replicated input

$\exists X.B$ output a type

1 unit for \otimes

0 unit for \oplus

X^\perp dual of type variable

$A \wp B$ input A then behave as B

$A \& B$ offer choice of A or B

$?A$ replicated output

$\forall X.B$ input a type

\perp unit for \wp

\top unit for $\&$

Duals

$$(X)^\perp = X^\perp$$

$$(X^\perp)^\perp = X$$

$$(A \otimes B)^\perp = A^\perp \wp B^\perp$$

$$(A \wp B)^\perp = A^\perp \otimes B^\perp$$

$$(A \oplus B)^\perp = A^\perp \& B^\perp$$

$$(A \& B)^\perp = A^\perp \oplus B^\perp$$

$$(!A)^\perp = ?A^\perp$$

$$(?A)^\perp = !A^\perp$$

$$(\exists X.B)^\perp = \forall X.B^\perp$$

$$(\forall X.B)^\perp = \exists X.B^\perp$$

$$1^\perp = \perp$$

$$\perp^\perp = 1$$

$$0^\perp = \top$$

$$\top^\perp = 0$$

Processes

$P, Q, R ::=$

$x \leftrightarrow y$	link	$\nu x : A.(P \mid Q)$	parallel composition
$x[y].(P \mid Q)$	output	$x(y).P$	input
$x[\text{inl}].P$	left selection	$x.\mathbf{case}(P, Q)$	choice
$x[\text{inr}].P$	right selection		
$?x[y].P$	replicated output	$!x(y).P$	replicated input
$x[A].P$	send type	$x(X).P$	receive type
$x[].0$	empty output	$x().P$	empty input
		$x.\mathbf{case}()$	empty choice

Forms $x(y).P$ and $!x(y).P$ behave like the same forms in π -calculus.

Forms $x[y].P$ and $?x[y].P$ behave like form $\nu y. x\langle y \rangle.P$ in π -calculus.

Structural rules

$$\frac{}{w \leftrightarrow x \vdash w : A^\perp, x : A} \text{Ax}$$

$$\frac{P \vdash \Gamma, x : A \quad Q \vdash \Delta, x : A^\perp}{\nu x : A. (P \mid Q) \vdash \Gamma, \Delta} \text{Cut}$$

(AxCut)

$$\frac{\frac{}{w \leftrightarrow x \vdash w : A^\perp, x : A} \text{Ax} \quad P \vdash \Gamma, x : A^\perp}{\nu x. (w \leftrightarrow x \mid P) \vdash \Gamma, w : A^\perp} \text{Cut} \quad \Longrightarrow \quad P\{w/x\} \vdash \Gamma, w : A^\perp$$

Structural rules—equivalences

(Swap)

$$\frac{P \vdash \Gamma, x : A \quad Q \vdash \Delta, x : A^\perp}{\nu x : A. (P \mid Q) \vdash \Gamma, \Delta} \text{Cut} \quad \equiv \quad \frac{Q \vdash \Delta, x : A^\perp \quad P \vdash \Gamma, x : A}{\nu x : A^\perp. (Q \mid P) \vdash \Gamma, \Delta} \text{Cut}$$

(Assoc)

$$\frac{\frac{P \vdash \Gamma, x : A \quad Q \vdash \Delta, x : A^\perp, y : B}{\nu x. (P \mid Q) \vdash \Gamma, \Delta, y : B} \text{Cut} \quad R \vdash \Theta, y : B^\perp}{\nu y. (\nu x. (P \mid Q) \mid R) \vdash \Gamma, \Delta, \Theta} \text{Cut} \quad \equiv \quad \frac{P \vdash \Gamma, x : A \quad \frac{Q \vdash \Delta, x : A^\perp, y : B \quad R \vdash \Theta, y : B^\perp}{\nu y. (Q \mid R) \vdash \Delta, \Theta, x : A^\perp} \text{Cut}}{\nu x. (P \mid \nu y. (Q \mid R)) \vdash \Gamma, \Delta, \Theta} \text{Cut}$$

Input and Output—Multiplicatives

$$\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{x[y].(P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes$$

$$\frac{R \vdash \Theta, y : A, x : B}{x(y).R \vdash \Theta, x : A \wp B} \wp$$

$(\beta_{\otimes \wp})$

$$\frac{\frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B}{x[y].(P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes \quad \frac{R \vdash \Theta, y : A^\perp, x : B^\perp}{x(y).R \vdash \Theta, x : A^\perp \wp B^\perp} \wp}{\nu x.(x[y].(P \mid Q) \mid x(y).R) \vdash \Gamma, \Delta, \Theta} \text{Cut} \quad \Longrightarrow$$

$$\frac{P \vdash \Gamma, y : A \quad \frac{Q \vdash \Delta, x : B \quad R \vdash \Theta, y : A^\perp, x : B^\perp}{\nu x.(Q \mid R) \vdash \Delta, \Theta, y : A^\perp} \text{Cut}}{\nu y.(P \mid \nu x.(Q \mid R)) \vdash \Gamma, \Delta, \Theta} \text{Cut}$$

Input and Output—identity, swap

$$\frac{\frac{\overline{y \leftrightarrow x \vdash y : A^\perp, x : A} \text{Ax}}{\overline{z[x].(w \leftrightarrow z \mid y \leftrightarrow x) \vdash y : A^\perp, w : B^\perp, z : A \otimes B} \otimes} \text{Ax}}{\overline{w(y).z[x].(w \leftrightarrow z \mid y \leftrightarrow x) \vdash w : A^\perp \wp B^\perp, z : A \otimes B} \wp}$$

$$\frac{\frac{\overline{w \leftrightarrow z \vdash w : B^\perp, z : B} \text{Ax}}{\overline{x[z].(w \leftrightarrow z \mid y \leftrightarrow x) \vdash w : B^\perp, y : A^\perp, x : B \otimes A} \otimes} \text{Ax}}{\overline{w(y).x[z].(w \leftrightarrow z \mid y \leftrightarrow x) \vdash w : A^\perp \wp B^\perp, x : B \otimes A} \wp}$$

Selection and Choice—Additives

$$\frac{P \vdash \Gamma, x : A}{x[\text{inl}].P \vdash \Gamma, x : A \oplus B} \oplus_1 \quad \frac{P \vdash \Gamma, x : B}{x[\text{inr}].P \vdash \Gamma, x : A \oplus B} \oplus_2$$

$$\frac{Q \vdash \Delta, x : A \quad R \vdash \Delta, x : B}{x.\text{case}(Q, R) \vdash \Delta, x : A \& B} \&$$

$(\beta_{\oplus \&})$

$$\frac{\frac{P \vdash \Gamma, x : A}{x[\text{inl}].P \vdash \Gamma, x : A \oplus B} \oplus_1 \quad \frac{Q \vdash \Delta, x : A^\perp \quad R \vdash \Delta, x : B^\perp}{x.\text{case}(Q, R) \vdash \Delta, x : A^\perp \& B^\perp} \&}{\nu x.(x[\text{inl}].P \mid x.\text{case}(Q, R)) \vdash \Gamma, \Delta} \text{Cut} \implies$$

$$\frac{P \vdash \Gamma, x : A \quad Q \vdash \Delta, x : A^\perp}{\nu x.(P \mid Q) \vdash \Gamma, \Delta} \text{Cut}$$

Servers and Clients—Exponentials

$$\frac{P \vdash ?\Gamma, y : A}{!x(y).P \vdash ?\Gamma, x : !A} !$$

$$\frac{Q \vdash \Delta, y : A}{?x[y].Q \vdash \Delta, x : ?A} ?$$

($\beta_{!?}$)

$$\frac{\frac{P \vdash ?\Gamma, y : A}{!x(y).P \vdash ?\Gamma, x : !A} ! \quad \frac{Q \vdash \Delta, y : A^\perp}{?x[y].Q \vdash \Delta, x : ?A^\perp} ?}{\nu x.(!x(y).P \mid ?x[y].Q) \vdash ?\Gamma, \Delta} ? \text{Cut} \implies$$

$$\frac{P \vdash ?\Gamma, y : A \quad Q \vdash \Delta, y : A^\perp}{\nu y.(P \mid Q) \vdash ?\Gamma, \Delta} \text{Cut}$$

Weakening and Contraction

$$\frac{Q \vdash \Delta}{Q \vdash \Delta, x : ?A} \text{ Weaken}$$

$$\frac{Q \vdash \Delta, x : ?A, x' : ?A}{Q\{x/x'\} \vdash \Delta, x : ?A} \text{ Contract}$$

$(\beta!W)$

$$\frac{\frac{P \vdash ?\Gamma, y : A}{!x(y).P \vdash ?\Gamma, x : !A} \quad \frac{Q \vdash \Delta}{Q \vdash \Delta, x : ?A^\perp}}{\nu x.(!x(y).P \mid Q) \vdash ?\Gamma, \Delta} \begin{array}{l} \text{Weaken} \\ \text{Cut} \end{array} \implies$$

$$\frac{Q \vdash \Delta}{Q \vdash ?\Gamma, \Delta} \text{ Weaken}$$

Weakening and Contraction, continued

$(\beta!C)$

$$\frac{\frac{P \vdash ?\Gamma, y : A}{!x(y).P \vdash ?\Gamma, x : !A} \quad ! \quad \frac{Q \vdash \Delta, x : ?A, x' : ?A}{Q\{x/x'\} \vdash \Delta, x : ?A}}{\nu x.(!x(y).P \mid Q\{x/x'\}) \vdash ?\Gamma, \Delta} \text{Contract} \quad \text{Cut} \quad \Longrightarrow$$

$$\frac{\frac{P \vdash ?\Gamma, y : A}{!x(y).P \vdash ?\Gamma, x : !A} \quad ! \quad \frac{\frac{P' \vdash ?\Gamma', y' : A}{!x'(y').P' \vdash ?\Gamma', x' : !A} \quad ! \quad Q \vdash \Delta, x : ?A^\perp, x' : ?A^\perp}{\nu x'.(!x'(y').P' \mid Q) \vdash ?\Gamma', \Delta, x : ?A^\perp} \text{Cut}}{\nu x.(!x(y).P \mid \nu x'.(!x'(y').P' \mid Q)) \vdash ?\Gamma, ?\Gamma', \Delta} \text{Cut} \quad \text{Contract}$$

Polymorphism—Quantifiers

$$\frac{P \vdash \Gamma, x : B\{A/X\}}{x[A].P \vdash \Gamma, x : \exists X.B} \exists$$

$$\frac{Q \vdash \Delta, x : B}{x(X).Q \vdash \Delta, x : \forall X.B} \forall \quad (X \notin \text{fv}(\Delta))$$

$$\begin{array}{c} (\beta_{\exists\forall}) \\ \frac{\frac{P \vdash \Gamma, x : B\{A/X\}}{x[A].P \vdash \Gamma, x : \exists X.B} \exists \quad \frac{Q \vdash \Delta, x : B^\perp}{x(X).Q \vdash \Delta, x : \forall X.B^\perp} \forall}{\nu x.(x[A].P \mid x(X).Q) \vdash \Gamma, \Delta} \text{Cut} \end{array} \implies$$

$$\frac{P \vdash \Gamma, x : B\{A/X\} \quad Q\{A/X\} \vdash \Delta, x : B^\perp\{A/X\}}{\nu x.(P \mid Q\{A/X\}) \vdash \Gamma, \Delta} \text{Cut}$$

Units

$$\frac{}{x[].0 \vdash x : 1} \mathbf{1} \quad \frac{P \vdash \Gamma}{x().P \vdash \Gamma, x : \perp} \perp$$

$$\text{(no rule for 0)} \quad \frac{}{x.\text{case}() \vdash \Gamma, x : \top} \top$$

$$\begin{array}{c} (\beta_{1\perp}) \\ \frac{\frac{}{x[].0 \vdash x : 1} \mathbf{1} \quad \frac{P \vdash \Gamma}{x().P \vdash \Gamma, x : \perp} \perp}{\nu x.(x[].0 \mid x().P) \vdash \Gamma} \text{Cut} \quad \perp}{\Rightarrow} P \vdash \Gamma \end{array}$$

$$(\beta_{0\top})$$

(no rule for 0 with \top)

Commuting Conversions

$$\begin{array}{c}
 (\kappa_{\otimes 1}) \\
 \frac{P \vdash \Gamma, y : A, z : C \quad Q \vdash \Delta, x : B}{x[y].(P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes \quad R \vdash \Theta, z : C^\perp \\
 \hline
 \frac{\nu z.(x[y].(P \mid Q) \mid R) \vdash \Gamma, \Delta, \Theta, x : A \otimes B}{\text{Cut}} \implies
 \end{array}$$

$$\begin{array}{c}
 \frac{P \vdash \Gamma, y : A, z : C \quad R \vdash \Theta, z : C^\perp}{\nu z.(P \mid R) \vdash \Gamma, \Theta, y : A} \text{Cut} \quad Q \vdash \Delta, x : B \\
 \hline
 \frac{x[y].(\nu z.(P \mid R) \mid Q) \vdash \Gamma, \Delta, \Theta, x : A \otimes B}{\otimes}
 \end{array}$$

$$\begin{array}{c}
 (\kappa_{\otimes 2}) \\
 \frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B, z : C}{x[y].(P \mid Q) \vdash \Gamma, \Delta, x : A \otimes B} \otimes \quad R \vdash \Theta, z : C^\perp \\
 \hline
 \frac{\nu z.(x[y].(P \mid Q) \mid R) \vdash \Gamma, \Delta, \Theta, x : A \otimes B}{\text{Cut}} \implies
 \end{array}$$

$$\begin{array}{c}
 \frac{P \vdash \Gamma, y : A \quad Q \vdash \Delta, x : B, z : C \quad R \vdash \Theta, z : C^\perp}{\nu z.(Q \mid R) \vdash \Delta, \Theta, x : B} \text{Cut} \\
 \hline
 \frac{x[y].(P \mid \nu z.(Q \mid R)) \vdash \Gamma, \Delta, \Theta, x : A \otimes B}{\otimes}
 \end{array}$$

Commuting Conversions

$$(\kappa_{\otimes 1}) \quad \nu z.(x[y].(P \mid Q) \mid R) \Longrightarrow x[y].(\nu z.(P \mid R) \mid Q), \quad \text{if } z \in \text{fn}(P)$$

$$(\kappa_{\otimes 2}) \quad \nu z.(x[y].(P \mid Q) \mid R) \Longrightarrow x[y].(P \mid \nu z.(Q \mid R)), \quad \text{if } z \in \text{fn}(Q)$$

$$(\kappa_{\wp}) \quad \nu z.(x(y).P \mid Q) \Longrightarrow x(y).\nu z.(P \mid Q)$$

$$(\kappa_{\oplus}) \quad \nu z.(x[\text{inl}].P \mid Q) \Longrightarrow x[\text{inl}].\nu z.(P \mid Q)$$

$$(\kappa_{\&}) \quad \nu z.(x.\text{case}(P, Q) \mid R) \Longrightarrow x.\text{case}(\nu z.(P \mid R), \nu z.(Q \mid R))$$

$$(\kappa_{!}) \quad \nu z.(!x(y).P \mid Q) \Longrightarrow !x(y).\nu z.(P \mid Q)$$

$$(\kappa_{?}) \quad \nu z.(?x[y].P \mid Q) \Longrightarrow ?x[y].\nu z.(P \mid Q)$$

$$(\kappa_{\exists}) \quad \nu z.(x[A].P \mid Q) \Longrightarrow x[A].\nu z.(P \mid Q)$$

$$(\kappa_{\forall}) \quad \nu z.(x(X).P \mid Q) \Longrightarrow x(X).\nu z.(P \mid Q)$$

$$(\kappa_{\perp}) \quad \nu z.(x().P \mid Q) \Longrightarrow x().\nu z.(P \mid Q)$$

$$(\kappa_{\top}) \quad \nu z.(x.\text{case}() \mid Q) \Longrightarrow x.\text{case}()$$

No congruence!

If our goal was to eliminate all cuts, we would need to introduce congruence rules, such as

$$\frac{P \Longrightarrow Q}{x(y).P \Longrightarrow x(y).Q}$$

and similarly for each operator. Such rules do not correspond well to our notion of computation on processes, so we omit them; this is analogous to the usual practice of not permitting reduction under lambda.

Cut Elimination

Theorem

(Subject Reduction)

If $P \vdash \Gamma$ and $P \Longrightarrow Q$ then $Q \vdash \Gamma$.

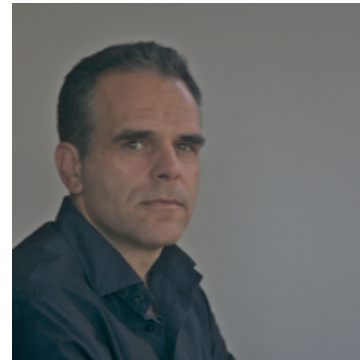
(Cut Elimination)

If $P \vdash \Gamma$ then there exists a Q
such that $P \Longrightarrow^* Q$ and Q is not a Cut.

Part II

GV

Good Variation Gay-Vasconcelos



Type Preservation

Theorem

(Translation preserves types)

If $\Phi \vdash M : T$

then $\llbracket M \rrbracket z \vdash \llbracket \Phi \rrbracket^\perp, z : \llbracket T \rrbracket$.

Session Types

$S ::=$

$!T.S$ output value of type T then behave as S

$?T.S$ input value of type T then behave as S

$\oplus\{l_i : S_i\}_{i \in I}$ select from behaviours S_i with label l_i

$\&\{l_i : S_i\}_{i \in I}$ offer choice of behaviours S_i with label l_i

$\text{end}_!$ terminator, convenient for use with output

$\text{end}_?$ terminator, convenient for use with input

Each session S has a dual \bar{S} :

$$\begin{array}{lcl} \overline{!T.S} & = & ?T.\bar{S} \\ \overline{\oplus(l_i : S_i)_{i \in I}} & = & \&(l_i : \bar{S}_i)_{i \in I} \\ \overline{\text{end}_!} & = & \text{end}_? \end{array} \qquad \begin{array}{lcl} \overline{?T.S} & = & !T.\bar{S} \\ \overline{\&(l_i : S_i)_{i \in I}} & = & \oplus(l_i : \bar{S}_i)_{i \in I} \\ \overline{\text{end}_?} & = & \text{end}_! \end{array}$$

Types

$T, U, V ::=$

S session (linear)

$T \otimes U$ tensor product (linear)

$T \multimap U$ function (linear)

$T \rightarrow U$ function (unlimited)

Unit unit (unlimited)

Each type is classified as linear or unlimited:

$\text{lin}(S)$ $\text{lin}(T \otimes U)$ $\text{lin}(T \multimap U)$

$\text{un}(T \rightarrow U)$ $\text{un}(\mathbf{Unit})$

Terms

$L, M, N ::=$

x

identifier

unit

unit constant

$\lambda x. N$

function abstraction

$L M$

function application

(M, N)

pair construction

let $(x, y) = M$ in N

pair deconstruction

send M N

send value M on channel N

receive M

receive from channel M

select l M

select label l on channel M

case M of $\{l_i : x.N_i\}_{i \in I}$

offer choice on channel M

with x connect M to N

connect M to N by channel x

terminate M

terminate input

Functions and Pairs

$$\frac{}{x : T \vdash x : T} \text{Id} \quad \frac{}{\vdash \text{unit} : \text{Unit}} \text{Unit}$$

$$\frac{\Phi \vdash N : U \quad \text{un}(T)}{\Phi, x : T \vdash N : U} \text{Weaken} \quad \frac{\Phi, x : T, x' : T \vdash N : U \quad \text{un}(T)}{\Phi, x : T \vdash N\{x/x'\} : U} \text{Contract}$$

$$\frac{\Phi, x : T \vdash N : U}{\Phi \vdash \lambda x. N : T \multimap U} \multimap\text{-I} \quad \frac{\Phi \vdash L : T \multimap U \quad \Psi \vdash M : T}{\Phi, \Psi \vdash LM : U} \multimap\text{-E}$$

$$\frac{\Phi \vdash L : T \multimap U \quad \text{un}(\Phi)}{\Phi \vdash L : T \rightarrow U} \rightarrow\text{-I} \quad \frac{\Phi \vdash L : T \rightarrow U}{\Phi \vdash L : T \multimap U} \rightarrow\text{-E}$$

$$\frac{\Phi \vdash M : T \quad \Psi \vdash N : U}{\Phi, \Psi \vdash (M, N) : T \otimes U} \otimes\text{-I} \quad \frac{\Phi \vdash M : T \otimes U \quad \Psi, x : T, y : U \vdash N : V}{\Phi, \Psi \vdash \text{let } (x, y) = M \text{ in } N : V} \otimes\text{-E}$$

Communication

$$\frac{\Phi \vdash M : T \quad \Psi \vdash N : !T.S}{\Phi, \Psi \vdash \text{send } M N : S} \text{ Send} \qquad \frac{\Phi \vdash M : ?T.S}{\Phi \vdash \text{receive } M : T \otimes S} \text{ Receive}$$

$$\frac{\Phi \vdash M : \oplus \{l_i : S_i\}_{i \in I}}{\Phi \vdash \text{select } l_j M : S_j} \text{ Select}$$

$$\frac{\Phi \vdash M : \& \{l_i : S_i\}_{i \in I} \quad (\Psi, x : S_i \vdash N_i : T)_{i \in I}}{\Phi, \Psi \vdash \text{case } M \text{ of } \{l_i : x.N_i\}_{i \in I} : T} \text{ Case}$$

$$\frac{\Phi, x : S \vdash M : \text{end}_! \quad \Psi, x : \bar{S} \vdash N : T}{\Phi, \Psi \vdash \text{with } x \text{ connect } M \text{ to } N : T} \text{ Connect}$$

$$\frac{\Phi \vdash M : T \otimes \text{end}_?}{\Phi \vdash \text{terminate } M : T} \text{ Terminate}$$

Translation of Sessions

$$\llbracket !T.S \rrbracket = \llbracket T \rrbracket^\perp \wp \llbracket S \rrbracket$$

$$\llbracket ?T.S \rrbracket = \llbracket T \rrbracket \otimes \llbracket S \rrbracket$$

$$\llbracket \oplus \{l_i : S_i\}_{i \in I} \rrbracket = \llbracket S_1 \rrbracket \& \cdots \& \llbracket S_n \rrbracket, \quad I = \{1, \dots, n\}$$

$$\llbracket \& \{l_i : S_i\}_{i \in I} \rrbracket = \llbracket S_1 \rrbracket \oplus \cdots \oplus \llbracket S_n \rrbracket, \quad I = \{1, \dots, n\}$$

$$\llbracket \text{end}_! \rrbracket = \perp$$

$$\llbracket \text{end}_? \rrbracket = 1$$

Translation preserves duality:

$$\llbracket \bar{S} \rrbracket = \llbracket S \rrbracket^\perp$$

Translation of Types

$$\llbracket T \multimap U \rrbracket = \llbracket T \rrbracket^\perp \wp \llbracket U \rrbracket$$

$$\llbracket T \rightarrow U \rrbracket = !(\llbracket T \rrbracket^\perp \wp \llbracket U \rrbracket)$$

$$\llbracket T \otimes U \rrbracket = \llbracket T \rrbracket \otimes \llbracket U \rrbracket$$

$$\llbracket \text{Unit} \rrbracket = !\top$$

An unlimited type translates to a type constructed with !:

If $\text{un}(T)$ then $\llbracket T \rrbracket = !A$, for some A .

Translation of Linear Functions

$$\left[\frac{\Phi, x : T \vdash N : U}{\Phi \vdash \lambda x. N : T \multimap U} \multimap\text{-I} \right] z = \frac{[N]z \vdash [\Phi]^\perp, x : [T]^\perp, z : [U]}{z(x).[N]z \vdash [\Phi]^\perp, z : [T]^\perp \wp [U]} \wp$$

$$\left[\frac{\Phi \vdash L : T \multimap U \quad \Psi \vdash M : T}{\Phi, \Psi \vdash LM : U} \multimap\text{-E} \right] z =$$

$$\frac{[L]y \vdash [\Phi]^\perp, y : [T]^\perp \wp [U] \quad \frac{[M]x \vdash [\Psi]^\perp, x : [T] \quad y \leftrightarrow z \vdash y : [U]^\perp, z : [U]}{y[x].([M]x \mid y \leftrightarrow z) \vdash [\Psi]^\perp, y : [T] \otimes [U]^\perp, z : [U]} \otimes}{\nu y.([L]y \mid y[x].([M]x \mid y \leftrightarrow z)) \vdash [\Phi]^\perp, [\Psi]^\perp, z : [U]} \text{Cut}$$

Translation of Unlimited Functions

$$\left[\frac{\Phi \vdash L : T \multimap U \quad \text{un}(\Phi)}{\Phi \vdash L : T \rightarrow U} \rightarrow\text{-I} \right] z =$$

$$\frac{[L]y \vdash [\Phi]^\perp, y : [T \multimap U]}{!z(y).[L]y \vdash [\Phi]^\perp, z : ![T \multimap U]} !$$

$$\left[\frac{\Phi \vdash L : T \rightarrow U}{\Phi \vdash L : T \multimap U} \rightarrow\text{-E} \right] z =$$

$$\frac{[L]y \vdash [\Phi]^\perp, y : ![T \multimap U] \quad \frac{x \leftrightarrow z \vdash x : [T \multimap U]^\perp, z : [T \multimap U] \quad \text{Ax}}{?y[x].x \leftrightarrow z \vdash y : ?[T \multimap U]^\perp, z : [T \multimap U]} ?}{\nu y.([L]y \mid ?y[x].x \leftrightarrow z) \vdash [\Phi]^\perp, z : [T \multimap U]} \text{Cut}$$

Translation of Send and Receive

$$\left[\frac{\Phi \vdash M : T \quad \Psi \vdash N : !T.S}{\Phi, \Psi \vdash \text{send } M N : S} \text{ Send} \right] z =$$

$$\frac{\frac{[[M]]y \vdash [[\Phi]]^\perp, y : [[T]] \quad x \leftrightarrow z \vdash x : [[S]]^\perp, z : [[S]]}{x[y].([M]y \mid x \leftrightarrow z) \vdash [[\Phi]]^\perp, x : [[T]] \otimes [[S]]^\perp} \otimes \quad \frac{[[N]]x \vdash [[\Psi]]^\perp, x : [[T]]^\perp \wp [[S]]}{\nu x.(x[y].([M]y \mid x \leftrightarrow z) \mid [[N]]x) \vdash [[\Phi]]^\perp, [[\Psi]]^\perp, z : [[S]]} \text{ Cut}}{}$$

$$\left[\frac{\Phi \vdash M : ?T.S}{\Phi \vdash \text{receive } M : T \otimes S} \text{ Receive} \right] z =$$

$$[[M]]z \vdash [[\Phi]], z : [[T]] \otimes [[S]]$$

Translation of Connect and Terminate

$$\left[\frac{\Phi, x : S \vdash M : \text{end}_! \quad \Psi, x : \bar{S} \vdash N : T}{\Phi, \Psi \vdash \text{with } x \text{ connect } M \text{ to } N : T} \text{Connect} \right] z =$$

$$\frac{\frac{[M]y \vdash [\Phi]^\perp, x : [S]^\perp, y : \perp \quad \overline{y[] . 0} \vdash y : 1}{\nu y. ([M]y \mid y[] . 0) \vdash [\Phi]^\perp, x : [S]^\perp} \quad 1}{\nu x. (\nu y. ([M]y \mid y[] . 0) \mid [N]z) \vdash [\Phi]^\perp, [\Psi]^\perp, z : [T]} \text{Cut} \quad [N]z \vdash [\Psi]^\perp, x : [S], z : [T]}$$

$$\left[\frac{\Phi \vdash M : T \otimes \text{end}_?}{\Phi \vdash \text{terminate } M : T} \text{Terminate} \right] z =$$

$$\frac{[M]y \vdash [\Phi]^\perp, y : [T] \otimes 1 \quad \frac{\frac{z \leftrightarrow y \vdash z : [T], y : [T]^\perp}{x().z \leftrightarrow y \vdash z : [T], y : [T]^\perp, x : \perp} \perp}{y(x).x().z \leftrightarrow y \vdash z : [T], y : [T]^\perp \wp \perp} \wp}{\nu y. ([M]y \mid y(x).x().z \leftrightarrow y) \vdash [\Phi]^\perp, z : [T]} \text{Cut}$$

Type Preservation

Theorem

(Translation preserves types)

If $\Phi \vdash M : T$

then $\llbracket M \rrbracket z \vdash \llbracket \Phi \rrbracket^\perp, z : \llbracket T \rrbracket$.

Part III

Conclusions and future work

Paradoxical combinator

$$X = X \supset A$$

$$\frac{\frac{\frac{[x : X \supset A]^x \quad [x : X]^x}{\supset\text{-E}} \quad \frac{[x : X \supset A]^x \quad [x : X]^x}{\supset\text{-E}}}{\frac{xx : A}{\lambda x. xx : X \supset A} \supset\text{-I}^x} \quad \frac{\frac{[x : X \supset A]^x \quad [x : X]^x}{\supset\text{-E}} \quad \frac{[x : X \supset A]^x \quad [x : X]^x}{\supset\text{-E}}}{\frac{xx : A}{\lambda x. xx : X} \supset\text{-I}^x} \supset\text{-E}}{\frac{(\lambda x. xx) (\lambda x. xx) : A}{\supset\text{-E}}}$$

Fixpoint combinator

$$X = X \supset A$$

$$\begin{array}{c}
 \frac{[f : A \supset A]^f \quad \frac{[x : X \supset A]^x \quad [x : X]^x}{x x : A} \supset\text{-E}}{f (x x) : A} \supset\text{-E} \quad \frac{[x : X \supset A]^x \quad [x : X]^x}{x x : A} \supset\text{-E}}{\frac{f (x x) : A}{\lambda x. f (x x) : X \supset A} \supset\text{-I}^x} \supset\text{-E} \\
 \frac{(\lambda x. f (x x)) (\lambda x. f (x x)) : A}{\lambda f. (\lambda x. f (x x)) (\lambda x. f (x x)) : (A \supset A) \supset A} \supset\text{-I}^f
 \end{array}$$

Restoring the full power of π -calculus

- Mix rule, Girard (1987)

$$\frac{P \vdash \Gamma \quad Q \vdash \Delta}{P \mid Q \vdash \Gamma, \Delta} \text{ Mix}$$

- Binary Cut rule, Abramsky, Gay, and Nagarajan (1996)

$$\frac{P \vdash \Gamma, x : A, y : B \quad Q \vdash \Delta, x : A^\perp, y : B^\perp}{\nu x : A, y : B. (P \mid Q) \vdash \Gamma, \Delta} \text{ BiCut}$$

