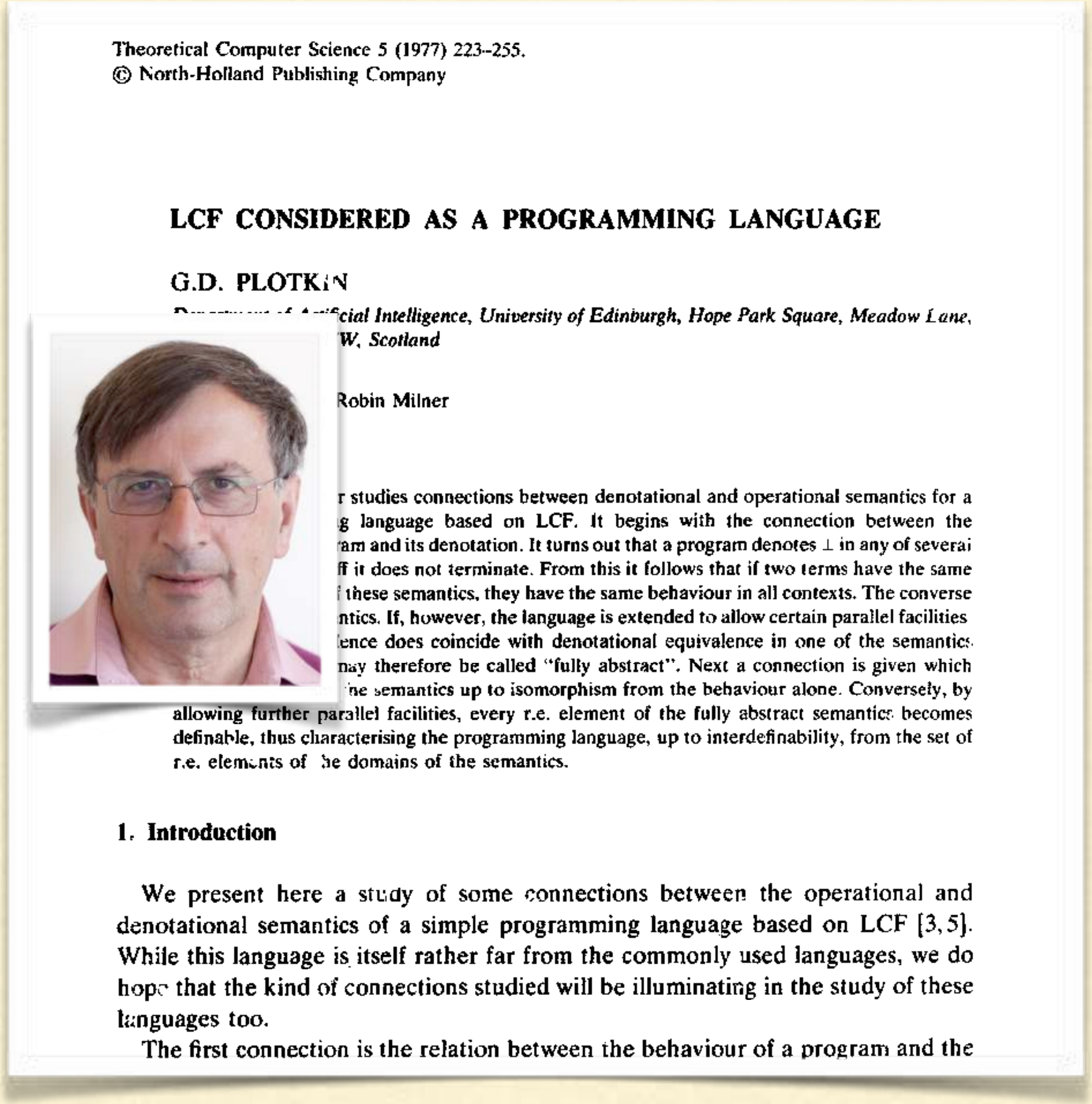
A PROBABILISTIC AND NON-DETERMINISTIC CALL-BY-PUSH-VALUE LANGUAGE

Jean Goubault-Larrecq

- PCF, probabilistic choice, and the trouble with **V**
- Curing the trouble using call-by-push-value
- Semantics, adequacy, full abstraction

PLOTKIN'S PCF (1977)



Theoretical Computer Science 5 (1977) 223–255.
© North-Holland Publishing Company

LCF CONSIDERED AS A PROGRAMMING LANGUAGE

G.D. PLOTKIN

…cial Intelligence, University of Edinburgh, Hope Park Square, Meadow Lane,
…W, Scotland

Robin Milner

…r studies connections between denotational and operational semantics for a …g language based on LCF. It begins with the connection between the …am and its denotation. It turns out that a program denotes ⊥ in any of several …f it does not terminate. From this it follows that if two terms have the same … these semantics, they have the same behaviour in all contexts. The converse …ntics. If, however, the language is extended to allow certain parallel facilities …ence does coincide with denotational equivalence in one of the semantics …ay therefore be called "fully abstract". Next a connection is given which …he semantics up to isomorphism from the behaviour alone. Conversely, by allowing further parallel facilities, every r.e. element of the fully abstract semantics becomes definable, thus characterising the programming language, up to interdefinability, from the set of r.e. elements of the domains of the semantics.

1. Introduction

We present here a study of some connections between the operational and denotational semantics of a simple programming language based on LCF [3,5]. While this language is itself rather far from the commonly used languages, we do hope that the kind of connections studied will be illuminating in the study of these languages too.

The first connection is the relation between the behaviour of a program and the

- Types $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau$
- Terms $M, N, \ldots ::= x_\tau$

$$\begin{array}{l} \mid MN \\ \mid \lambda\, x_\sigma \,.\, M \\ \mid \mathbf{rec}\; x_\sigma \,.\, M \\ \mid \underline{n} \\ \mid \mathbf{succ}\; M \\ \mid \mathbf{pred}\; M \\ \mid \mathbf{ifz}\; M\; N\; P \end{array}$$

- (All terms are typed. Call by name.)

PLOTKIN'S PCF (1977)

- Types $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau$
- Terms
$$\begin{aligned} M, N, \ldots ::=\ & x_\tau \\ \mid\ & MN \\ \mid\ & \lambda\, x_\sigma \,.\, M \\ \mid\ & \mathbf{rec}\ x_\sigma \,.\, M \\ \mid\ & \underline{n} \\ \mid\ & \mathbf{succ}\ M \\ \mid\ & \mathbf{pred}\ M \\ \mid\ & \mathbf{ifz}\ M\ N\ P \end{aligned}$$
- (All terms are typed. Call by name.)

- An **operational** semantics:
$$M \to^* N$$
- A **denotational** semantics:
$$[\![M]\!]$$
- **Adequacy**:
for every ground M : **int**,
$$[\![M]\!]=n \text{ iff } M \to^* \underline{n}$$

PLOTKIN'S PCF (1977)

- An **operational** semantics:

  $$M \to^* N$$

- A **denotational** semantics:

  $$[\![M]\!]$$

- **Adequacy**:
  for every ground M : **int**,

  $$[\![M]\!]=n \text{ iff } M \to^* \underline{n}$$

- **Contextual preordering**:
  $M \lesssim N$ iff
  for every context C : **int**,

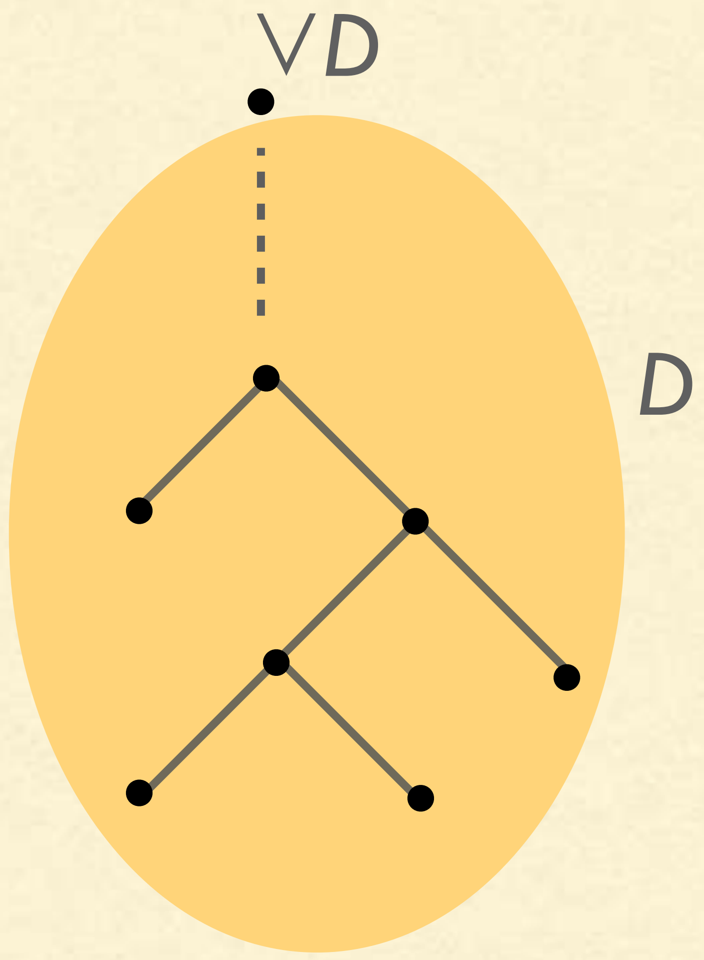
  $$C[M] \to^* \underline{n} \;\Rightarrow\; C[N] \to^* \underline{n}$$

- **Fact**: if $[\![M]\!] \leq [\![N]\!]$ then $M \lesssim N$

- Converse is **full abstraction**.
  Fails for PCF, works for PCF+**por**

DCPOS

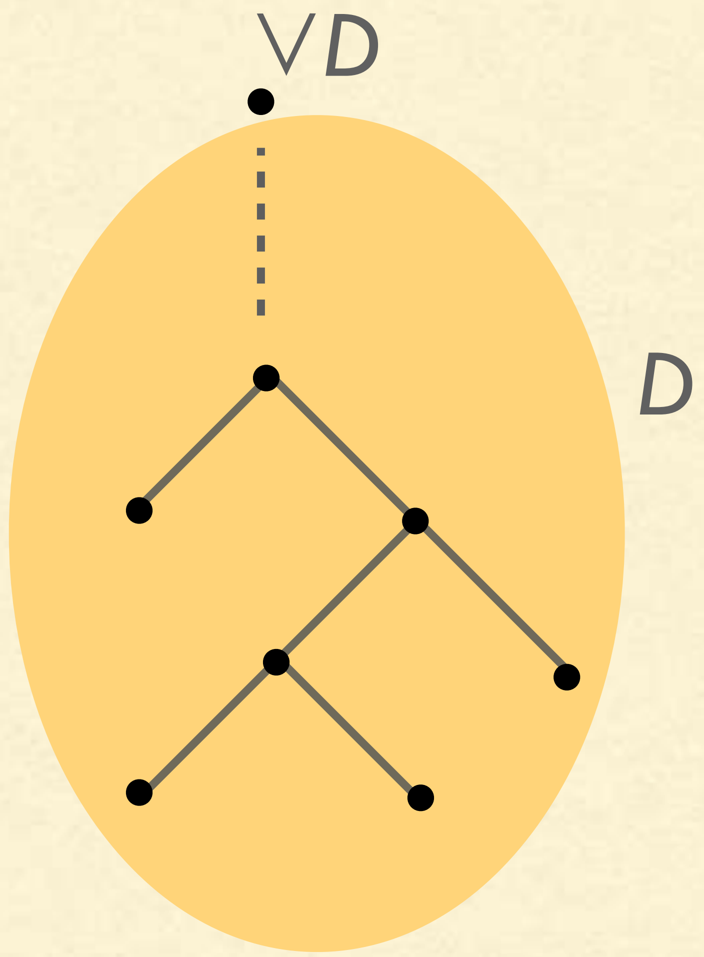
- Every type τ interpreted as a **dcpo** $[\![\tau]\!]$



A directed family D.
In a dcpo, every directed family D
has a supremum $\bigvee D$

DCPOS

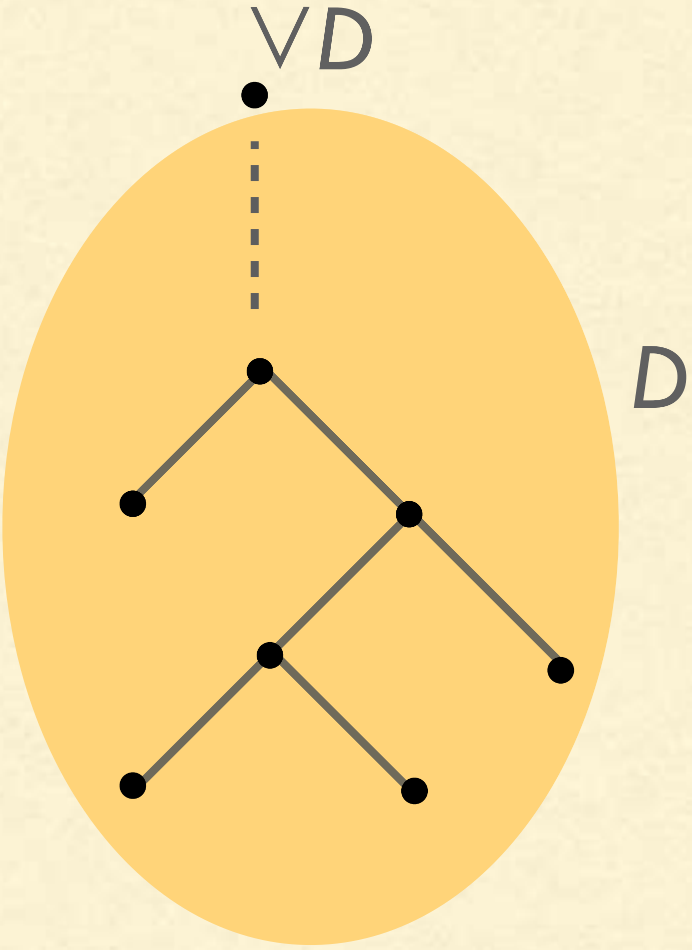
- Every type τ interpreted as a **dcpo** ⟦τ⟧
- ⟦**int**⟧ = $\mathbb{Z}_\perp$ ($\perp \leq n$, all n incomparable)



A directed family D.
In a dcpo, every directed family D has a supremum $\bigvee D$

DCPOS

- Every type τ interpreted as a **dcpo** $[\![\tau]\!]$
- $[\![$**int**$]\!] = \mathbb{Z}_\perp$ ($\perp \leq n$, all n incomparable)
- $[\![\sigma \to \tau]\!] = [[\![\sigma]\!] \to [\![\tau]\!]]$,
  dcpo of Scott-continuous maps : $[\![\sigma]\!] \to [\![\tau]\!]$
  (monotonic + preserves directed sups)



A directed family D.
In a dcpo, every directed family D
has a supremum $\bigvee D$

THE SEMANTICS OF PCF

- Types $\sigma, \tau, \ldots ::=$ **int** $\mid \sigma \to \tau$
- Terms $M, N, \ldots ::= x_\tau$
  $\mid MN$
  $\mid \lambda x_\sigma . M$
  $\mid$ **rec** $x_\sigma . M$
  $\mid \underline{n}$
  $\mid$ **succ** M
  $\mid$ **pred** M
  $\mid$ **ifz** $M\ N\ P$

- $[\![MN]\!] = \overbrace{[\![M]\!]}^{\in [\![\sigma \to \tau]\!]}(\overbrace{[\![N]\!]}^{\in [\![\sigma]\!]})$
  $[\![\lambda x_\sigma . M]\!] = (\underbrace{V}_{\in [\![\sigma]\!]} \mapsto \underbrace{[\![M]\!][x_\sigma{:=}V]}_{\in [\![\tau]\!]})$
- Meaningful since **Dcpo** is a Cartesian-closed category

CARTESIAN-CLOSEDNESS

- $[\![MN]\!] = \overbrace{[\![M]\!]}^{\in [\![\sigma \to \tau]\!]}(\overbrace{[\![N]\!]}^{\in [\![\sigma]\!]})$
  $[\![\lambda\, x_\sigma \,.\, M]\!] = (\underbrace{V}_{\in [\![\sigma]\!]} \mapsto \underbrace{[\![M]\!][x_\sigma := V]}_{\in [\![\tau]\!]})$
- Meaningful since **Dcpo** is a Cartesian-closed category

- In order to prove full abstraction (with por), we require to be able to **approximate** elements of $[\![\tau]\!]$ by **definable** elements $[\![M]\!]$.
- In the case of PCF, each $[\![\tau]\!]$ is an **algebraic bc-domain**, making that possible.
- Cartesian-closed… good.

CCCS OF CONTINUOUS DCPOS

- In order to prove full abstraction (with por), we require to be able to **approximate** elements of $\llbracket \tau \rrbracket$ by **definable** elements $\llbracket M \rrbracket$.

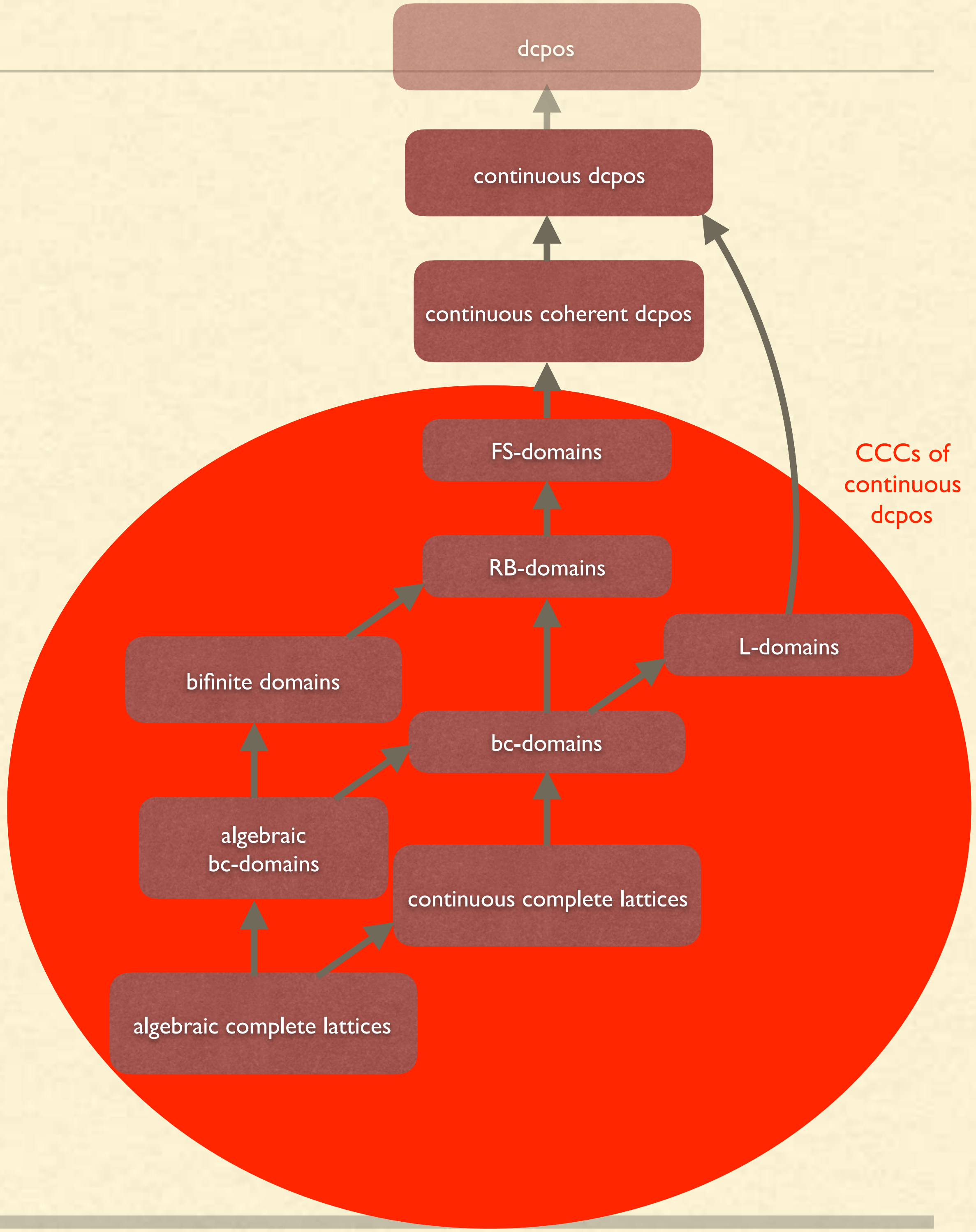
- In the case of PCF, each $\llbracket \tau \rrbracket$ is an **algebraic bc-domain**, making that possible.

- Cartesian-closed… good.

algebraic
bc-domains

CCCS OF CONTINUOUS DCPOS

- In order to prove full abstraction (with por), we require to be able to **approximate** elements of ⟦τ⟧ by **definable** elements ⟦M⟧.
- In the case of PCF, each ⟦τ⟧ is an **algebraic bc-domain**, making that possible.
- Cartesian-closed… good.
- Many other CCCs would fit, provided they consist of **continuous dcpos**.



ADDING PROBABILITIES

- Types
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau \mid \mathbf{V}\tau$

- Terms $M, N, \ldots ::= \ldots$
  $\mid M \oplus N$
  $\mid \mathbf{ret}\ M$
  $\mid \mathbf{do}\ x_\sigma \leftarrow M; N$

ADDING PROBABILITIES

Monadic type of subprobability valuations over τ

- Types
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau \mid \mathbf{V}\tau$

- Terms $M, N, \ldots ::= \ldots$
  $\mid M \oplus N$
  $\mid \mathbf{ret}\ M$
  $\mid \mathbf{do}\ x_\sigma \leftarrow M; N$

ADDING PROBABILITIES

- Types
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau \mid \mathbf{V}\tau$

- Terms $M, N, \ldots ::= \ldots$
  $\mid M \oplus N$
  $\mid \mathbf{ret}\ M$
  $\mid \mathbf{do}\ x_\sigma \leftarrow M; N$

Monadic type of subprobability valuations over τ

with M, N: $\mathbf{V}\tau$,
choose between M and N
with probability 1/2

ADDING PROBABILITIES

- Types
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \sigma \to \tau \mid \mathbf{V}\tau$

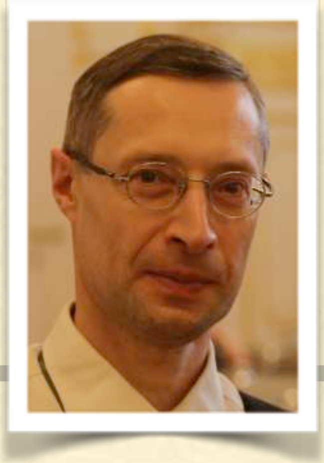
Monadic type of subprobability valuations over τ

- Terms $M, N, \ldots ::= \ldots$
  $\mid M \oplus N$
  $\mid \mathbf{ret}\ M$
  $\mid \mathbf{do}\ x_\sigma \leftarrow M; N$

with $M, N : \mathbf{V}\tau$,
choose between M and N
with probability 1/2

monadic constructions:
$M{:}\tau \Rightarrow \mathbf{ret}\ M{:}\mathbf{V}\tau$

$M{:}\mathbf{V}\sigma \quad N{:}\mathbf{V}\tau \Rightarrow \mathbf{do}\ x_\sigma \leftarrow M; N : \mathbf{V}\tau$

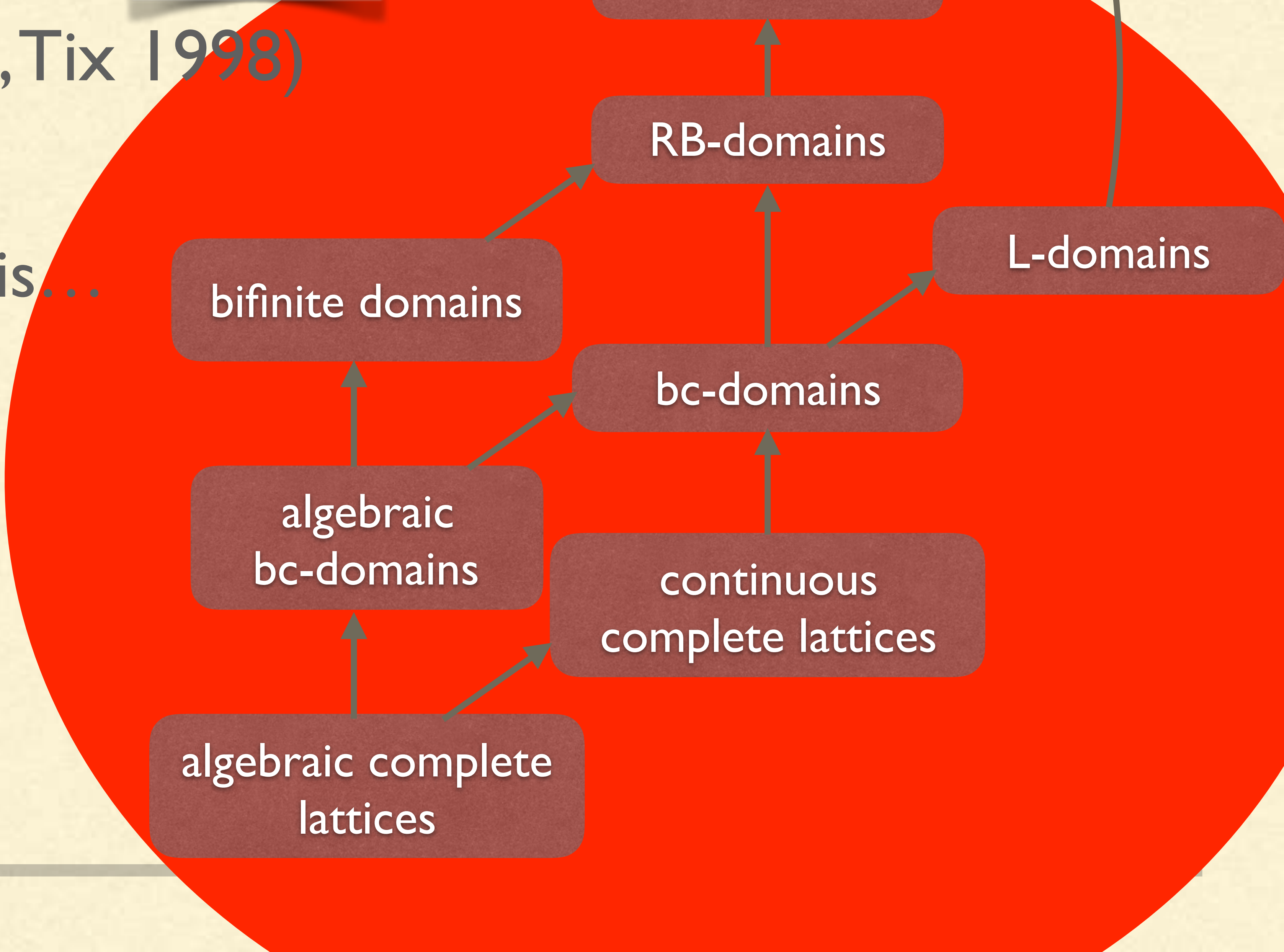


(Moggi 1991)

THE TROUBLE WITH **V**

(Jung, Tix 1998)

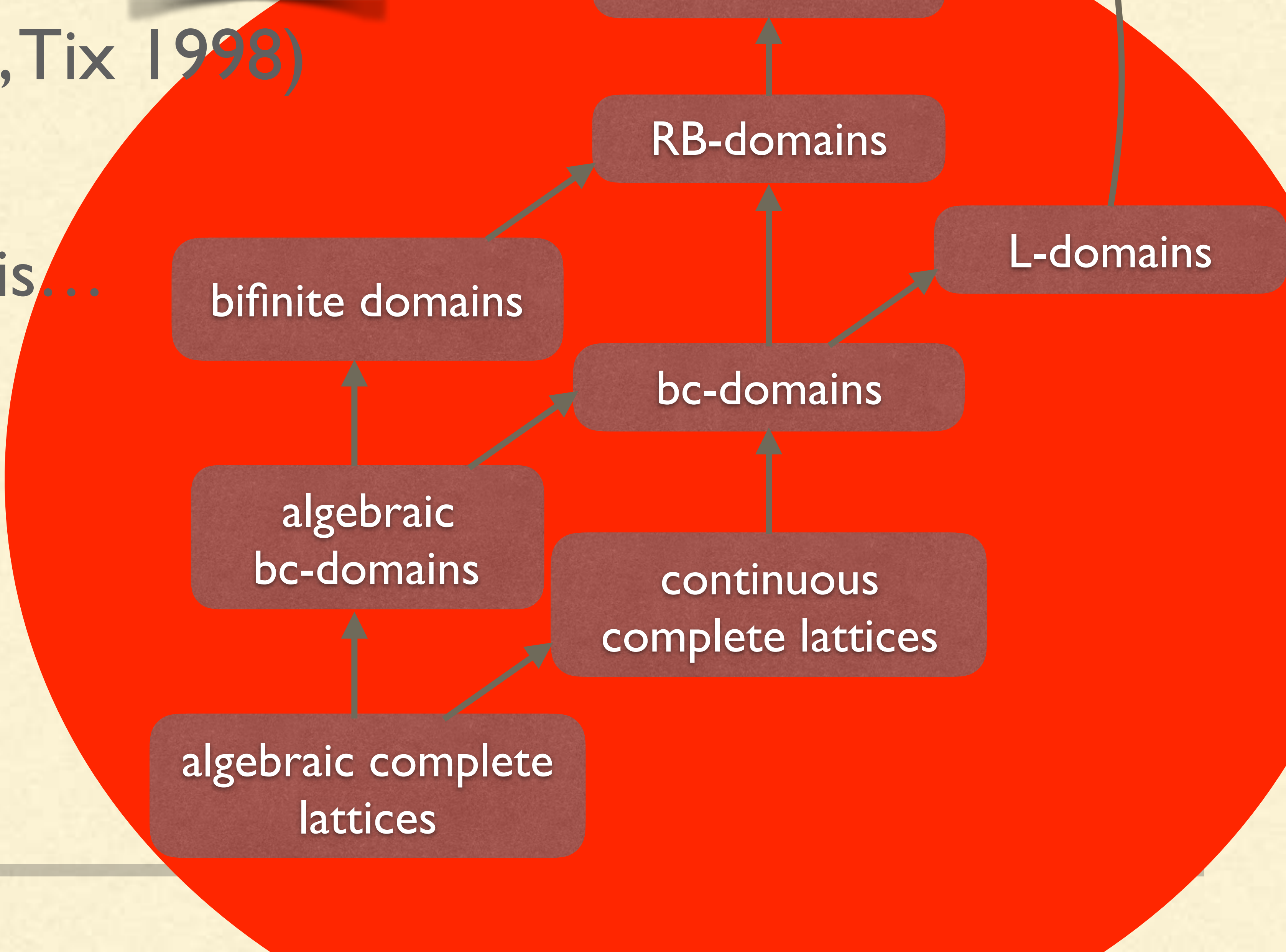
- Look for a category of continuous dcpos that is…



THE TROUBLE WITH **V**

(Jung, Tix 1998)

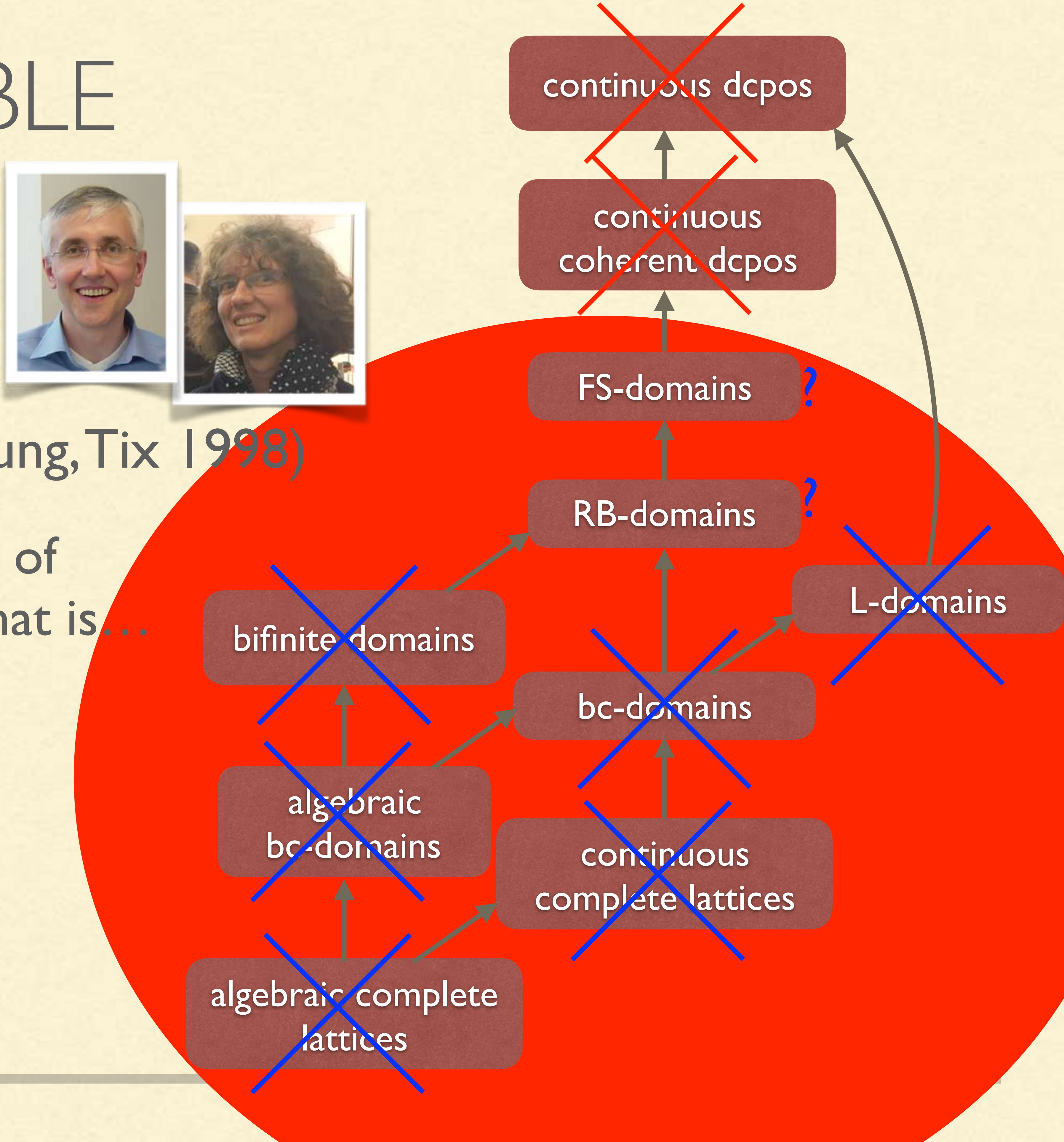
- Look for a category of continuous dcpos that is…
- Cartesian-closed



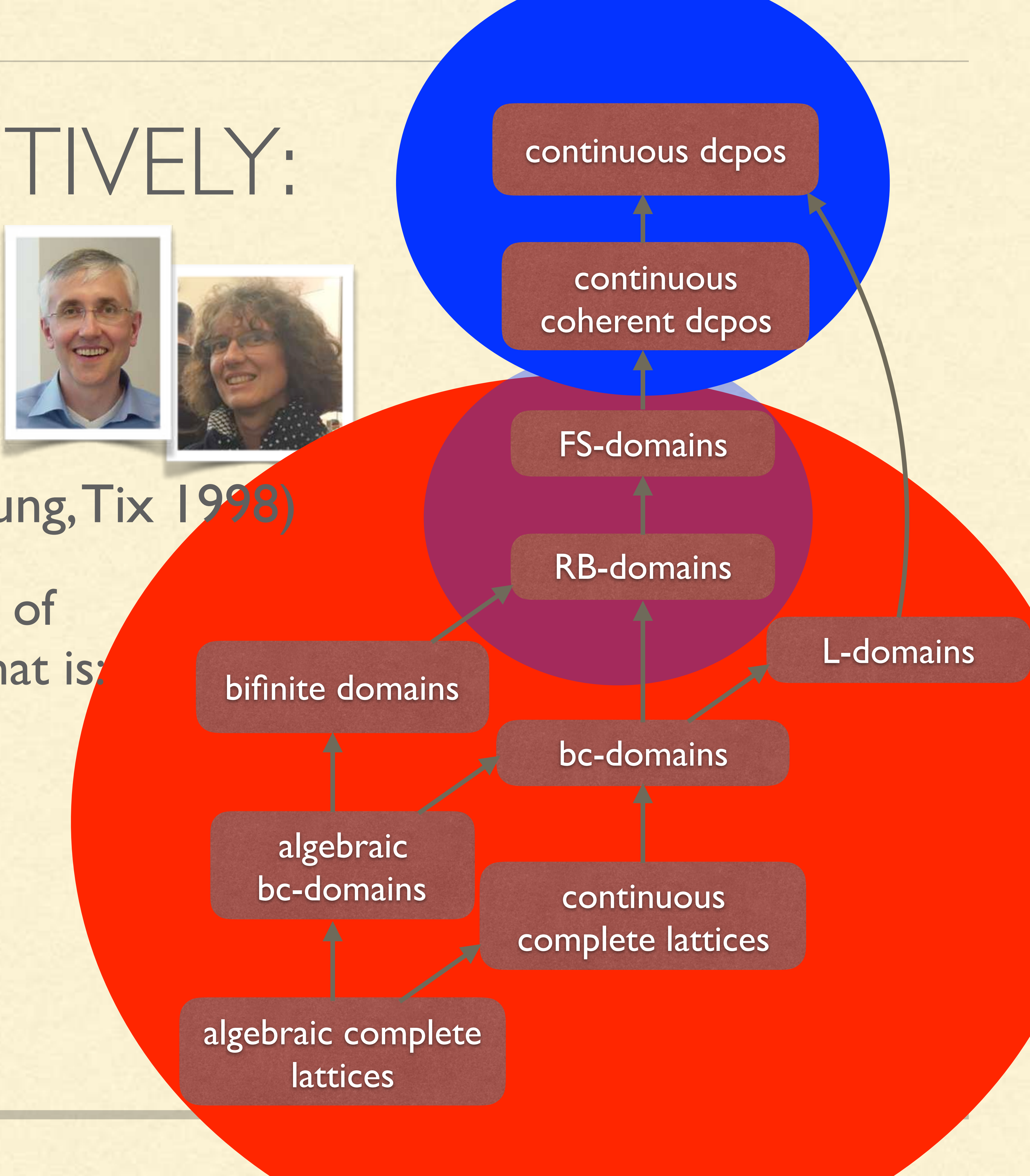
THE TROUBLE WITH **V**

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is…
- Cartesian-closed
- closed under **V**



MORE POSITIVELY:



(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **∨**

continuous dcpos

continuous coherent dcpos

FS-domains

RB-domains

L-domains

bifinite domains

bc-domains

algebraic bc-domains

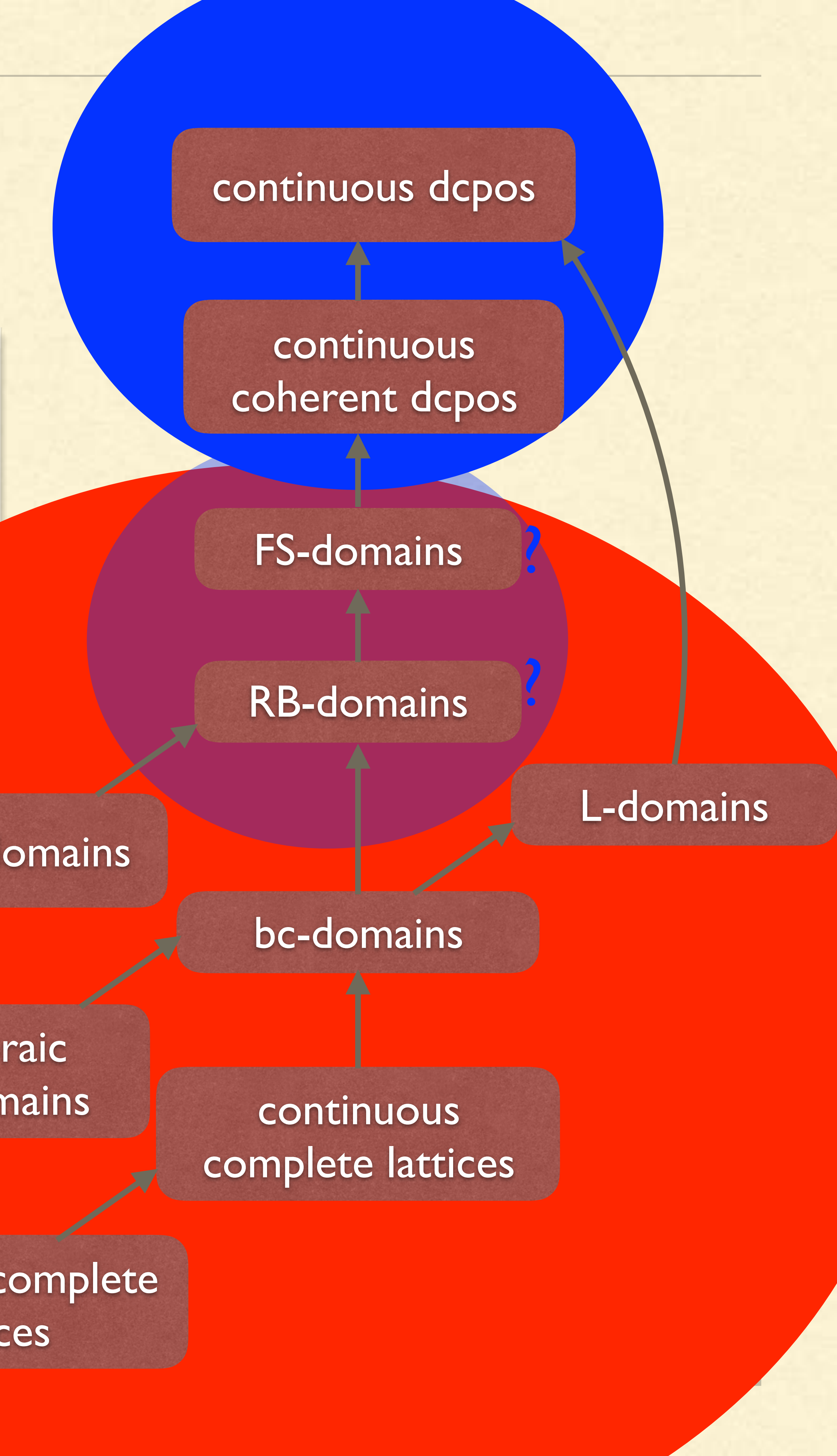
continuous complete lattices

algebraic complete lattices

MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **V**



OTHER SOLUTIONS (1)

- Change categories entirely. E.g., reason in **probabilistic coherence spaces**
- Equationally **fully abstract** semantics (Ehrhard, Pagani, Tasson 14)
- also for call-by-push-value (Ehrhard, Tasson 19)
- probabilistic choice ‘built-in’

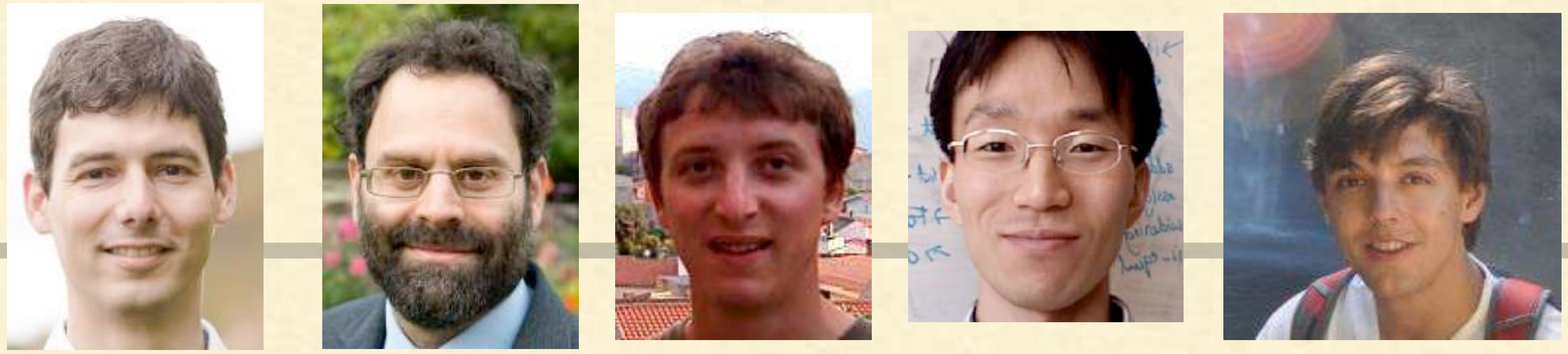
OTHER SOLUTIONS (2)

- Change categories, and opt for **QCB spaces/predomains** (Battenfeld 06)
  … Cartesian-closed, and has a probabilistic choice monad

OTHER SOLUTIONS (2)

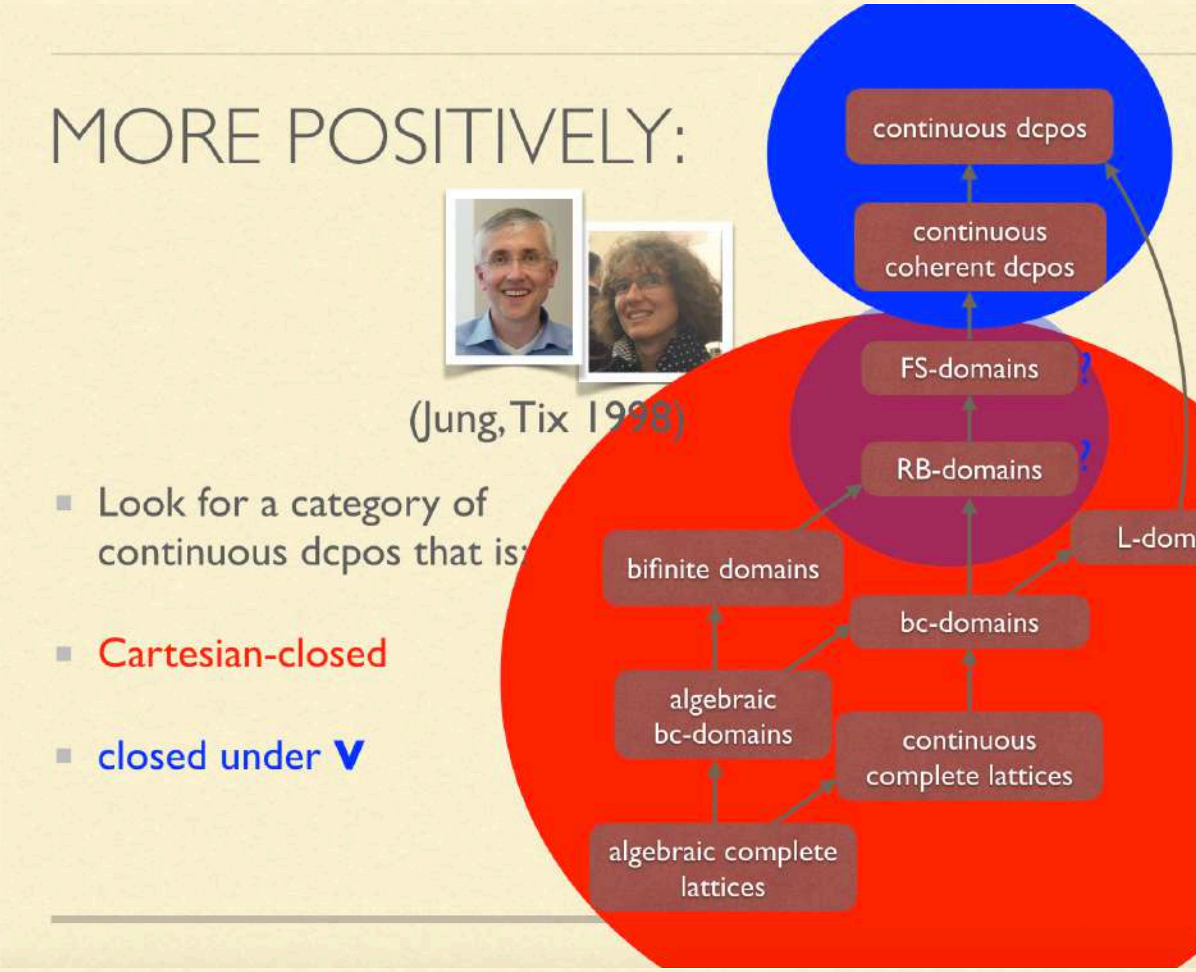
- Change categories, and opt for **QCB spaces/predomains** (Battenfeld 06)
  … Cartesian-closed, and has a probabilistic choice monad

- Changes categories, and opt for **quasi-Borel spaces/ domains**
  (Heunen, Kammar, Staton, Yang 17; Vákár, Kammar, Staton 19)
  … Cartesian-closed,
  and closed under a 'laws of random variables' functor



BACK TO DOMAINS

- There is no need to leave domain theory after all
- An easy solution using **call-by-push-value**
- will also handle the mix with **demonic non-determinism**



MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **V**

- PCF, probabilistic choice, and the trouble with **V**
- Curing the trouble using call-by-push-value
- Semantics, adequacy, full abstraction

- PCF, probabilistic choice, and the trouble with **V**
- Curing the trouble using call-by-push-value
- Semantics, adequacy, full abstraction

TWO KINDS OF TYPES?

- No such problem with two kinds of types:
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \ldots \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \ldots \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

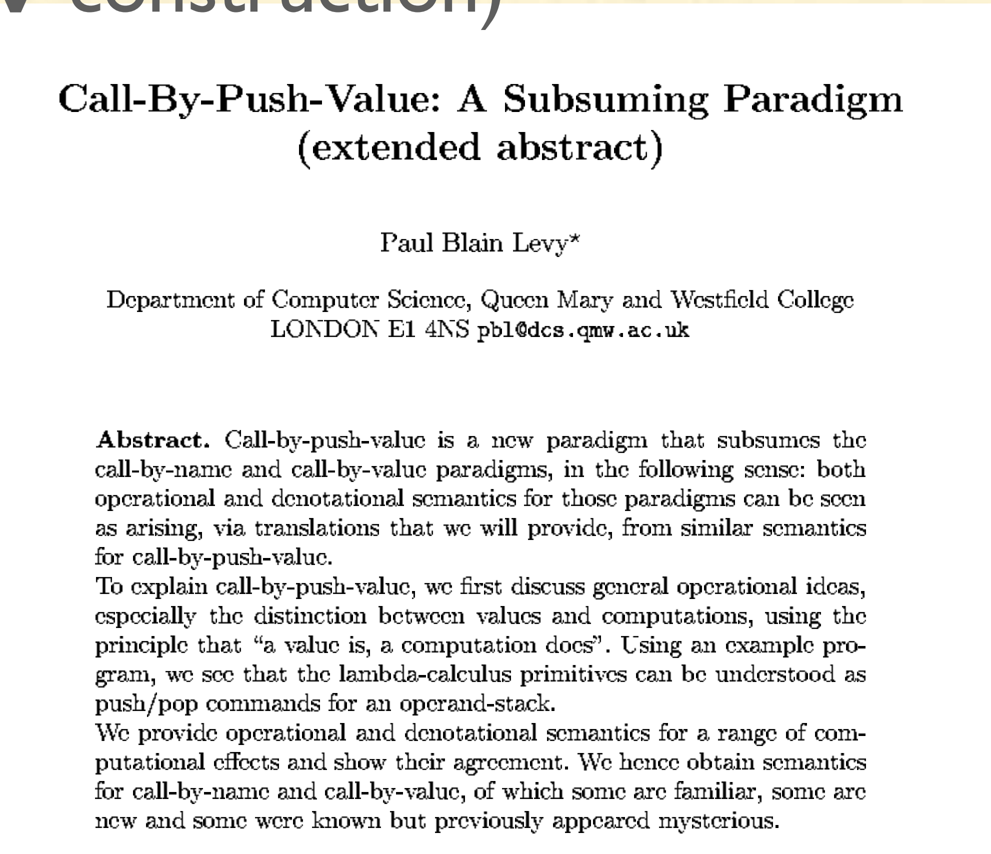
CALL-BY-PUSH-VALUE

- No such problem with two kinds of types:
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

- This is the type structure of Paul B. Levy's **call-by-push-value** (except for the **V** construction)



Call-By-Push-Value: A Subsuming Paradigm
(extended abstract)

Paul Blain Levy*

Department of Computer Science, Queen Mary and Westfield College
LONDON E1 4NS `pbl@dcs.qmw.ac.uk`

Abstract. Call-by-push-value is a new paradigm that subsumes the call-by-name and call-by-value paradigms, in the following sense: both operational and denotational semantics for those paradigms can be seen as arising, via translations that we will provide, from similar semantics for call-by-push-value.
To explain call-by-push-value, we first discuss general operational ideas, especially the distinction between values and computations, using the principle that "a value is, a computation does". Using an example program, we see that the lambda-calculus primitives can be understood as push/pop commands for an operand-stack.
We provide operational and denotational semantics for a range of computational effects and show their agreement. We hence obtain semantics for call-by-name and call-by-value, of which some are familiar, some are new and some were known but previously appeared mysterious.

(Levy 1999)

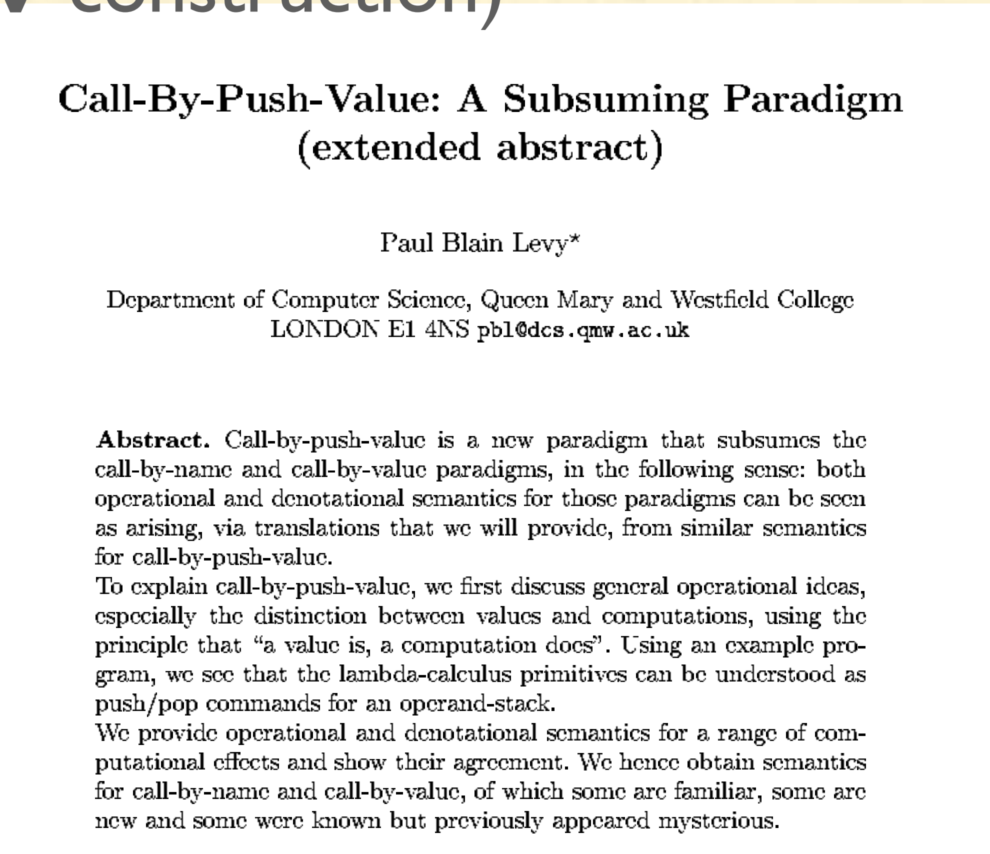
CALL-BY-PUSH-VALUE

- No such problem with two kinds of types:
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

value types

bc-domains/continuous lattices

- This is the type structure of Paul B. Levy's **call-by-push-value** (except for the **V** construction)



Call-By-Push-Value: A Subsuming Paradigm
(extended abstract)

Paul Blain Levy*

Department of Computer Science, Queen Mary and Westfield College
LONDON E1 4NS pbl@dcs.qmw.ac.uk

Abstract. Call-by-push-value is a new paradigm that subsumes the call-by-name and call-by-value paradigms, in the following sense: both operational and denotational semantics for those paradigms can be seen as arising, via translations that we will provide, from similar semantics for call-by-push-value.
To explain call-by-push-value, we first discuss general operational ideas, especially the distinction between values and computations, using the principle that "a value is, a computation does". Using an example program, we see that the lambda-calculus primitives can be understood as push/pop commands for an operand-stack.
We provide operational and denotational semantics for a range of computational effects and show their agreement. We hence obtain semantics for call-by-name and call-by-value, of which some are familiar, some are new and some were known but previously appeared mysterious.

(Levy 1999)

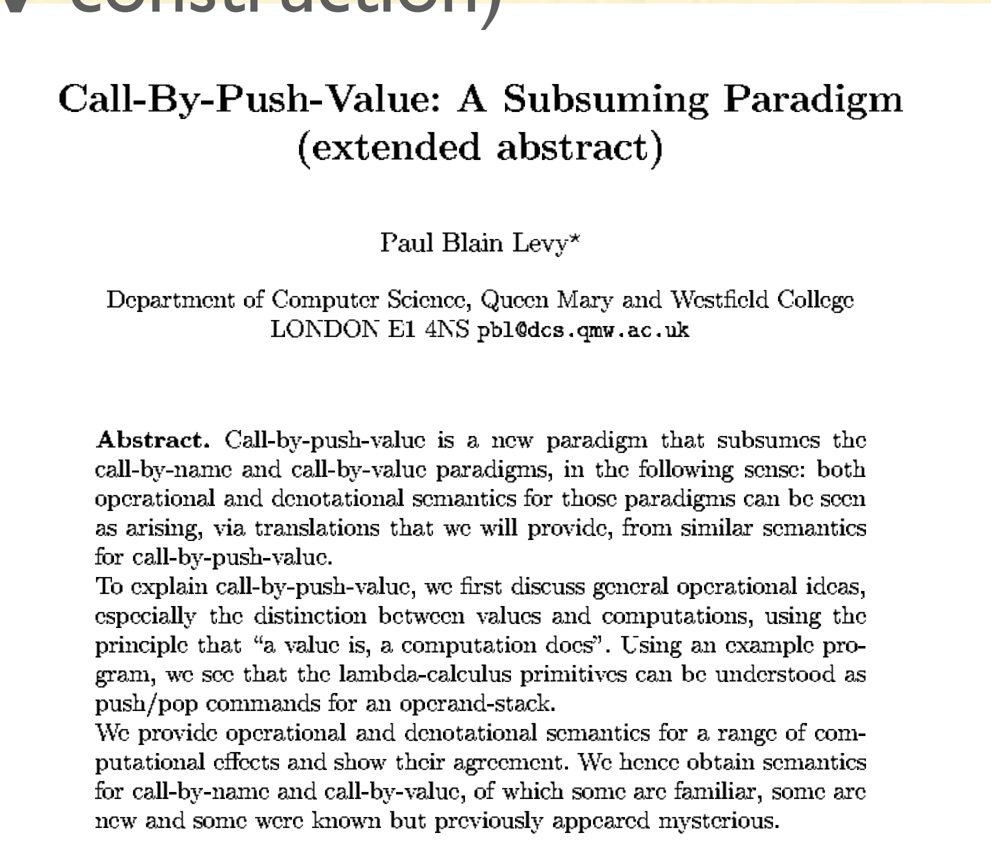
CALL-BY-PUSH-VALUE

- No such problem with two kinds of types:
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

value types

computation types

- This is the type structure of Paul B. Levy's **call-by-push-value** (except for the **V** construction)



Call-By-Push-Value: A Subsuming Paradigm (extended abstract)

Paul Blain Levy*

Department of Computer Science, Queen Mary and Westfield College
LONDON E1 4NS pbl@dcs.qmw.ac.uk

Abstract. Call-by-push-value is a new paradigm that subsumes the call-by-name and call-by-value paradigms, in the following sense: both operational and denotational semantics for those paradigms can be seen as arising, via translations that we will provide, from similar semantics for call-by-push-value.
To explain call-by-push-value, we first discuss general operational ideas, especially the distinction between values and computations, using the principle that "a value is, a computation does". Using an example program, we see that the lambda-calculus primitives can be understood as push/pop commands for an operand-stack.
We provide operational and denotational semantics for a range of computational effects and show their agreement. We hence obtain semantics for call-by-name and call-by-value, of which some are familiar, some are new and some were known but previously appeared mysterious.

(Levy 1999)

U AND **F**

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \sigma \times \tau \mid \mathbf{V}\tau$

  $\underline{\sigma}, \underline{\tau}, \ldots ::= \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

U AND **F**

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

- **U** converts from bc-domains to continuous coherent dcpos
  … semantically the identity: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$

U AND F

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

- **U** converts from bc-domains to continuous coherent dcpos
  … semantically the identity: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$

  - $M, N, \ldots ::= \ldots$
    | **force** *M* $(\mathbf{U}\underline{\sigma} \to \underline{\sigma})$
    | **thunk** *M* $(\underline{\sigma} \to \mathbf{U}\underline{\sigma})$

U AND **F**

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

- **U** converts from bc-domains to continuous coherent dcpos
  … semantically the identity: $\llbracket \mathbf{U}\underline{\sigma} \rrbracket = \llbracket \underline{\sigma} \rrbracket$

  - $M, N, \ldots ::= \ldots$
    $\mid$ **force** M $(\mathbf{U}\underline{\sigma} \to \underline{\sigma})$
    $\mid$ **thunk** M $(\underline{\sigma} \to \mathbf{U}\underline{\sigma})$

  - $\llbracket$**force** $M\rrbracket = \llbracket M \rrbracket$
    $\llbracket$**thunk** $M\rrbracket = \llbracket M \rrbracket$

  - **force thunk** $M \to M$

U AND F

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

bc-domains/continuous lattices

- **U** converts from bc-domains to continuous coherent dcpos
  … semantically the identity: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$

- **F** converts from continuous coherent dcpos to bc-domains
  … we take $[\![\mathbf{F}\sigma]\!]$=(lifted) Smyth powerdomain of $[\![\sigma]\!]$

THE SMYTH POWERDOMAIN

- **Q**X = {compact saturated subsets of X}, reverse inclusion $\supseteq$
- **Q**X is a continuous complete lattice
  for every continuous coherent dcpo X
- Serves as a model of **demonic non-determinism**.

THE SMYTH POWERDOMAIN

- $\mathbf{Q}X$ = {compact saturated subsets of X}, reverse inclusion $\supseteq$ defines a(nother) **monad** on the cat. of cont. coh. dcpos.
- **Unit:** $\eta : X \to \mathbf{Q}X : x \mapsto \uparrow x$ (continuous)
- **Extension:** for $f : X \to L$ where L continuous complete lattice,

$$\text{let } f^* : \mathbf{Q}X \to L : Q \mapsto \inf \{f(x) \mid x \in Q\}$$

— if f is continuous then f^* is continuous
— $f^* \circ \eta = f$
— $f^* \circ g^* = (f^* \circ g)^*$

THE SMYTH$_\perp$ POWERDOMAIN

- Technically, we use $\mathbf{Q}_\perp X$ = $\mathbf{Q}X$ plus a fresh bottom $\perp$
  … allows f^* to be **strict** now (needed for adequacy)

U AND F

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$

  continuous (coherent) dcpos

  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

  continuous complete lattices

- **U** converts from bc-domains to continuous coherent dcpos: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$

- **F** converts from continuous coherent dcpos to bc-domains: $[\![\mathbf{F}\sigma]\!]=\mathbf{Q}_{\perp}[\![\sigma]\!]$

U AND F

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

continuous complete lattices

- **U** converts from bc-domains to continuous coherent dcpos: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$
- **F** converts from continuous coherent dcpos to bc-domains: $[\![\mathbf{F}\sigma]\!]=\mathbf{Q}_\perp[\![\sigma]\!]$
  - $M, N, \ldots ::= \ldots$
    $\mid \mathbf{abort}_{\mathbf{F}\sigma}$
    $\mid M \owedge N$
    $\mid \mathbf{produce}\ M$ $\quad(\sigma \to \mathbf{F}\sigma)$
    $\mid M\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N$

choice

monad

U AND F

- $\sigma, \tau, \ldots ::= \textbf{int} \mid \textbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

continuous complete lattices

- **U** converts from bc-domains to continuous coherent dcpos: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$
- **F** converts from continuous coherent dcpos to bc-domains: $[\![\mathbf{F}\sigma]\!]=\mathbf{Q}_{\perp}[\![\sigma]\!]$
  - $M, N, \ldots ::= \ldots$
    $\mid \textbf{abort}_{\mathbf{F}\sigma}$
    $\mid M \owedge N$
    $\mid \textbf{produce}\ M \qquad (\sigma \to \mathbf{F}\sigma)$
    $\mid M\ \textbf{to}\ x_\sigma\ \textbf{in}\ N$
  - $[\![\textbf{abort}_{\mathbf{F}\sigma}]\!] = \varnothing$
    $[\![M \owedge N]\!] = [\![M]\!] \wedge [\![N]\!]$
    $[\![\textbf{produce}\ M]\!] = \eta([\![M]\!])$
    $[\![M\ \textbf{to}\ x_\sigma\ \textbf{in}\ N]\!] =$
    $(V \mapsto [\![N]\!][x_\sigma{:=}V])^* \, ([\![M]\!])$

choice

monad

U AND **F**

- $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$

continuous (coherent) dcpos

continuous complete lattices

- **U** converts from bc-domains to continuous coherent dcpos: $[\![\mathbf{U}\underline{\sigma}]\!]=[\![\underline{\sigma}]\!]$
- **F** converts from continuous coherent dcpos to bc-domains: $[\![\mathbf{F}\sigma]\!]=\mathbf{Q}_\perp[\![\sigma]\!]$
  - $M, N, \ldots ::= \ldots$
    $\mid \mathbf{abort}_{\mathbf{F}\sigma}$
    $\mid M \owedge N$
    $\mid \mathbf{produce}\ M \qquad (\sigma \to \mathbf{F}\sigma)$
    $\mid M\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N$

choice

monad

  - $[\![\mathbf{abort}_{\mathbf{F}\sigma}]\!] = \varnothing$
    $[\![M \owedge N]\!] = [\![M]\!] \wedge [\![N]\!]$
    $[\![\mathbf{produce}\ M]\!] = \eta([\![M]\!])$
    $[\![M\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N]\!] =$
    $(V \mapsto [\![N]\!][x_\sigma{:=}V])^*\ ([\![M]\!])$
- $(\mathbf{produce}\ M)\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N \to N[x_\sigma{:=}M]$ + etc.

- PCF, probabilistic choice, and the trouble with **V**
- Curing the trouble using call-by-push-value
- Semantics, adequacy, full abstraction

- PCF, probabilistic choice, and the trouble with **V**
- Curing the trouble using call-by-push-value
- Semantics, adequacy, full abstraction

OPERATIONAL SEMANTICS

- A Krivine machine for deterministic operations, working on configurations

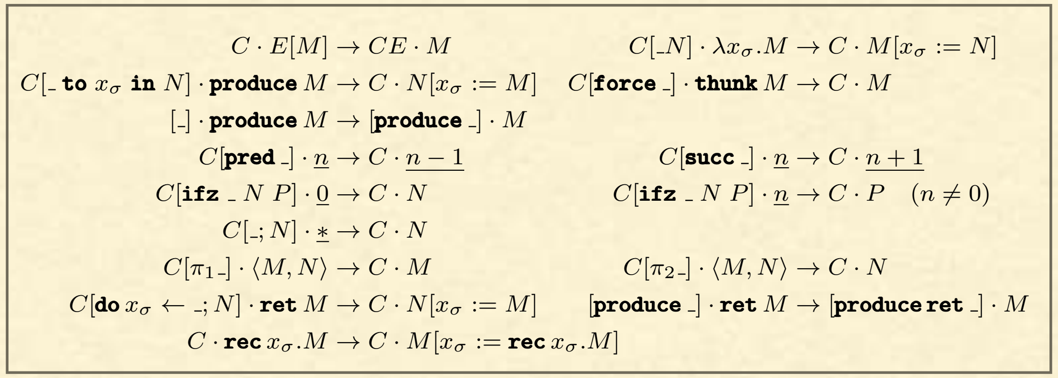
  C . M

$$\begin{array}{rr}
C \cdot E[M] \to CE \cdot M & C[\_N] \cdot \lambda x_\sigma . M \to C \cdot M[x_\sigma := N] \\
C[\_\ \texttt{to}\ x_\sigma\ \texttt{in}\ N] \cdot \texttt{produce}\ M \to C \cdot N[x_\sigma := M] & C[\texttt{force}\ \_] \cdot \texttt{thunk}\ M \to C \cdot M \\
[\_] \cdot \texttt{produce}\ M \to [\texttt{produce}\ \_] \cdot M & \\
C[\texttt{pred}\ \_] \cdot \underline{n} \to C \cdot \underline{n-1} & C[\texttt{succ}\ \_] \cdot \underline{n} \to C \cdot \underline{n+1} \\
C[\texttt{ifz}\ \_\ N\ P] \cdot \underline{0} \to C \cdot N & C[\texttt{ifz}\ \_\ N\ P] \cdot \underline{n} \to C \cdot P \quad (n \neq 0) \\
C[\_; N] \cdot \underline{*} \to C \cdot N & \\
C[\pi_1 \_] \cdot \langle M, N \rangle \to C \cdot M & C[\pi_2 \_] \cdot \langle M, N \rangle \to C \cdot N \\
C[\texttt{do}\ x_\sigma \leftarrow \_; N] \cdot \texttt{ret}\ M \to C \cdot N[x_\sigma := M] & [\texttt{produce}\ \_] \cdot \texttt{ret}\ M \to [\texttt{produce}\,\texttt{ret}\ \_] \cdot M \\
C \cdot \texttt{rec}\ x_\sigma . M \to C \cdot M[x_\sigma := \texttt{rec}\ x_\sigma . M] &
\end{array}$$

OPERATIONAL SEMANTICS

- A Krivine machine for deterministic operations, working on configurations

  $$C \,.\, M$$

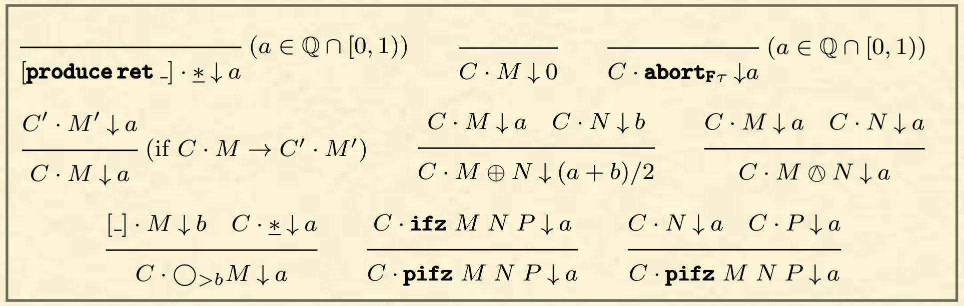


$$
\begin{array}{rr}
C \cdot E[M] \to CE \cdot M & C[\_N] \cdot \lambda x_\sigma.M \to C \cdot M[x_\sigma := N] \\
C[\_\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N] \cdot \mathbf{produce}\, M \to C \cdot N[x_\sigma := M] & C[\mathbf{force}\ \_] \cdot \mathbf{thunk}\, M \to C \cdot M \\
[\_] \cdot \mathbf{produce}\, M \to [\mathbf{produce}\ \_] \cdot M & \\
C[\mathbf{pred}\ \_] \cdot \underline{n} \to C \cdot \underline{n-1} & C[\mathbf{succ}\ \_] \cdot \underline{n} \to C \cdot \underline{n+1} \\
C[\mathbf{ifz}\ \_\ N\ P] \cdot \underline{0} \to C \cdot N & C[\mathbf{ifz}\ \_\ N\ P] \cdot \underline{n} \to C \cdot P \quad (n \neq 0) \\
C[\_; N] \cdot \underline{*} \to C \cdot N & \\
C[\pi_1\,\_] \cdot \langle M, N \rangle \to C \cdot M & C[\pi_2\,\_] \cdot \langle M, N \rangle \to C \cdot N \\
C[\mathbf{do}\, x_\sigma \leftarrow \_; N] \cdot \mathbf{ret}\, M \to C \cdot N[x_\sigma := M] & [\mathbf{produce}\ \_] \cdot \mathbf{ret}\, M \to [\mathbf{produce}\,\mathbf{ret}\ \_] \cdot M \\
C \cdot \mathbf{rec}\, x_\sigma.M \to C \cdot M[x_\sigma := \mathbf{rec}\, x_\sigma.M] &
\end{array}
$$

- Prob. must-termination judgments

  $$C \,.\, M \downarrow a$$

  (« whichever way you resolve the demonic non-deterministic choices, the probability that $C \,.\, M$ terminates is $>a$. »)



$$
\frac{}{[\mathbf{produce}\,\mathbf{ret}\ \_] \cdot \underline{*} \downarrow a}\ (a \in \mathbb{Q} \cap [0,1)) \qquad \frac{}{C \cdot M \downarrow 0} \qquad \frac{}{C \cdot \mathbf{abort}_{\mathbf{F}\tau} \downarrow a}\ (a \in \mathbb{Q} \cap [0,1))
$$

$$
\frac{C' \cdot M' \downarrow a}{C \cdot M \downarrow a}\ (\text{if } C \cdot M \to C' \cdot M') \qquad \frac{C \cdot M \downarrow a \quad C \cdot N \downarrow b}{C \cdot M \oplus N \downarrow (a+b)/2} \qquad \frac{C \cdot M \downarrow a \quad C \cdot N \downarrow a}{C \cdot M \owedge N \downarrow a}
$$

$$
\frac{[\_] \cdot M \downarrow b \quad C \cdot \underline{*} \downarrow a}{C \cdot \bigcirc_{>b} M \downarrow a} \qquad \frac{C \cdot \mathbf{ifz}\ M\ N\ P \downarrow a}{C \cdot \mathbf{pifz}\ M\ N\ P \downarrow a} \qquad \frac{C \cdot N \downarrow a \quad C \cdot P \downarrow a}{C \cdot \mathbf{pifz}\ M\ N\ P \downarrow a}
$$

OPERATIONAL SEMANTICS

- A Krivine machine for deterministic operations, working on configurations

  C . M

$$\begin{array}{rr}
C \cdot E[M] \to CE \cdot M & C[\_N] \cdot \lambda x_\sigma.M \to C \cdot M[x_\sigma := N] \\
C[\_\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N] \cdot \mathbf{produce}\, M \to C \cdot N[x_\sigma := M] & C[\mathbf{force}\ \_] \cdot \mathbf{thunk}\, M \to C \cdot M \\
[\_] \cdot \mathbf{produce}\, M \to [\mathbf{produce}\ \_] \cdot M & \\
C[\mathbf{pred}\ \_] \cdot \underline{n} \to C \cdot \underline{n-1} & C[\mathbf{succ}\ \_] \cdot \underline{n} \to C \cdot \underline{n+1} \\
C[\mathbf{ifz}\ \_\ N\ P] \cdot \underline{0} \to C \cdot N & C[\mathbf{ifz}\ \_\ N\ P] \cdot \underline{n} \to C \cdot P \quad (n \neq 0) \\
C[\_; N] \cdot \underline{*} \to C \cdot N & \\
C[\pi_1 \_] \cdot \langle M, N \rangle \to C \cdot M & C[\pi_2 \_] \cdot \langle M, N \rangle \to C \cdot N \\
C[\mathbf{do}\, x_\sigma \leftarrow \_; N] \cdot \mathbf{ret}\, M \to C \cdot N[x_\sigma := M] & [\mathbf{produce}\ \_] \cdot \mathbf{ret}\, M \to [\mathbf{produce}\,\mathbf{ret}\ \_] \cdot M \\
C \cdot \mathbf{rec}\, x_\sigma.M \to C \cdot M[x_\sigma := \mathbf{rec}\, x_\sigma.M] &
\end{array}$$

- Prob. must-termination judgments

  C . M ↓ a

  (« whichever way you resolve the demonic non-deterministic choices, the probability that *C . M* terminates is >*a.* »)

$$\frac{}{[\mathbf{produce}\,\mathbf{ret}\ \_] \cdot \underline{*} \downarrow a}\ (a \in \mathbb{Q} \cap [0,1)) \qquad \frac{}{C \cdot M \downarrow 0} \qquad \frac{}{C \cdot \mathbf{abort}_{\mathbf{F}\tau} \downarrow a}\ (a \in \mathbb{Q} \cap [0,1))$$

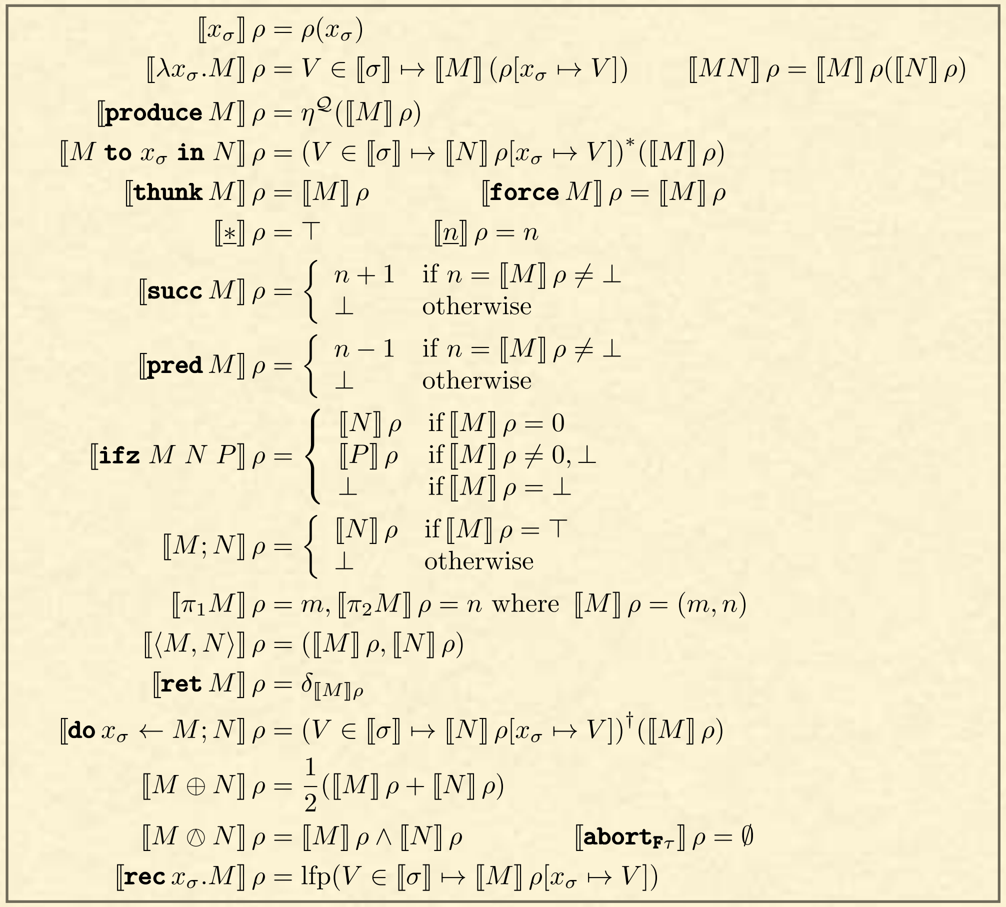
$$\frac{C' \cdot M' \downarrow a}{C \cdot M \downarrow a}\ (\text{if } C \cdot M \to C' \cdot M') \qquad \frac{C \cdot M \downarrow a \quad C \cdot N \downarrow b}{C \cdot M \oplus N \downarrow (a+b)/2} \qquad \frac{C \cdot M \downarrow a \quad C \cdot N \downarrow a}{C \cdot M \owedge N \downarrow a}$$

$$\frac{[\_] \cdot M \downarrow b \quad C \cdot \underline{*} \downarrow a}{C \cdot \bigcirc_{>b} M \downarrow a} \qquad \frac{C \cdot \mathbf{ifz}\ M\ N\ P \downarrow a}{C \cdot \mathbf{pifz}\ M\ N\ P \downarrow a} \qquad \frac{C \cdot N \downarrow a \quad C \cdot P \downarrow a}{C \cdot \mathbf{pifz}\ M\ N\ P \downarrow a}$$

- Let Pr(*C . M*↓) = sup {*a* | *C . M* ↓ *a*}, Pr(*M*↓) =Pr([_] . *M*↓)

ADEQUACY

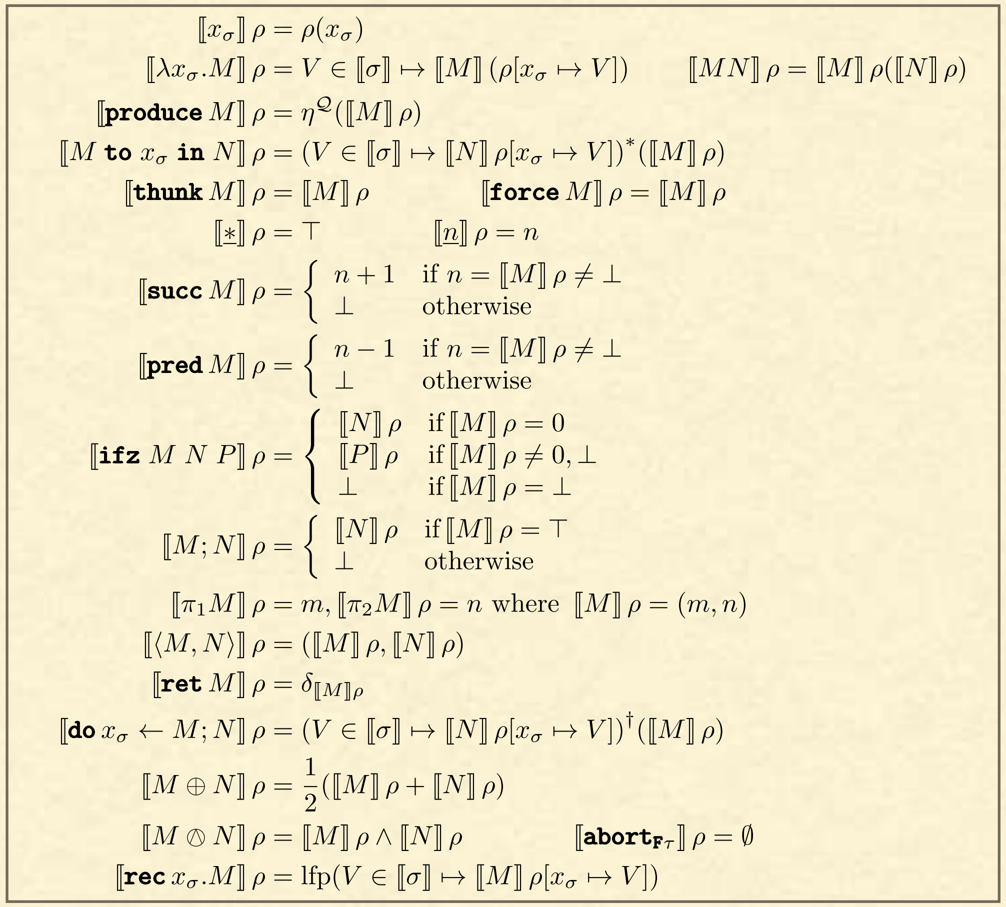
- **Prop (adequacy).**
  For every *M* : **FVunit**,
  — $[\![M]\!]=\perp$ and $\Pr(M\downarrow)=0$, or
  — $[\![M]\!]=\varnothing$ and $\Pr(M\downarrow)=1$, or else
  — $\Pr(M\downarrow)=\min\{\nu(\{\top\}) \mid \nu \in [\![M]\!]\}$



$$\begin{aligned}
[\![x_\sigma]\!]\,\rho &= \rho(x_\sigma) \\
[\![\lambda x_\sigma.M]\!]\,\rho &= V \in [\![\sigma]\!] \mapsto [\![M]\!]\,(\rho[x_\sigma \mapsto V]) \qquad [\![MN]\!]\,\rho = [\![M]\!]\,\rho([\![N]\!]\,\rho) \\
[\![\mathtt{produce}\,M]\!]\,\rho &= \eta^{\mathcal{Q}}([\![M]\!]\,\rho) \\
[\![M\ \mathtt{to}\ x_\sigma\ \mathtt{in}\ N]\!]\,\rho &= (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^*([\![M]\!]\,\rho) \\
[\![\mathtt{thunk}\,M]\!]\,\rho &= [\![M]\!]\,\rho \qquad [\![\mathtt{force}\,M]\!]\,\rho = [\![M]\!]\,\rho \\
[\![\underline{*}]\!]\,\rho &= \top \qquad [\![\underline{n}]\!]\,\rho = n \\
[\![\mathtt{succ}\,M]\!]\,\rho &= \begin{cases} n+1 & \text{if } n = [\![M]\!]\,\rho \neq \perp \\ \perp & \text{otherwise} \end{cases} \\
[\![\mathtt{pred}\,M]\!]\,\rho &= \begin{cases} n-1 & \text{if } n = [\![M]\!]\,\rho \neq \perp \\ \perp & \text{otherwise} \end{cases} \\
[\![\mathtt{ifz}\ M\ N\ P]\!]\,\rho &= \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = 0 \\ [\![P]\!]\,\rho & \text{if } [\![M]\!]\,\rho \neq 0, \perp \\ \perp & \text{if } [\![M]\!]\,\rho = \perp \end{cases} \\
[\![M;N]\!]\,\rho &= \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = \top \\ \perp & \text{otherwise} \end{cases} \\
[\![\pi_1 M]\!]\,\rho &= m, [\![\pi_2 M]\!]\,\rho = n \text{ where } [\![M]\!]\,\rho = (m,n) \\
[\![\langle M, N\rangle]\!]\,\rho &= ([\![M]\!]\,\rho, [\![N]\!]\,\rho) \\
[\![\mathtt{ret}\,M]\!]\,\rho &= \delta_{[\![M]\!]\rho} \\
[\![\mathtt{do}\,x_\sigma \leftarrow M;N]\!]\,\rho &= (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^\dagger([\![M]\!]\,\rho) \\
[\![M \oplus N]\!]\,\rho &= \frac{1}{2}([\![M]\!]\,\rho + [\![N]\!]\,\rho) \\
[\![M \owedge N]\!]\,\rho &= [\![M]\!]\,\rho \wedge [\![N]\!]\,\rho \qquad [\![\mathtt{abort}_{\mathtt{F}\tau}]\!]\,\rho = \emptyset \\
[\![\mathtt{rec}\,x_\sigma.M]\!]\,\rho &= \mathrm{lfp}(V \in [\![\sigma]\!] \mapsto [\![M]\!]\,\rho[x_\sigma \mapsto V])
\end{aligned}$$

ADEQUACY

- **Prop (adequacy).**
  For every M : **FVunit**,
  — $[\![M]\!]=\bot$ and $\Pr(M\!\downarrow)=0$, or
  — $[\![M]\!]=\varnothing$ and $\Pr(M\!\downarrow)=1$, or else
  — $\Pr(M\!\downarrow)=\min\{\nu(\{\top\}) \mid \nu \in [\![M]\!]\}$
- I.e., $\Pr(M\!\downarrow)=h^*([\![M]\!])$
  where $h(\nu) = \nu(\{\top\})$



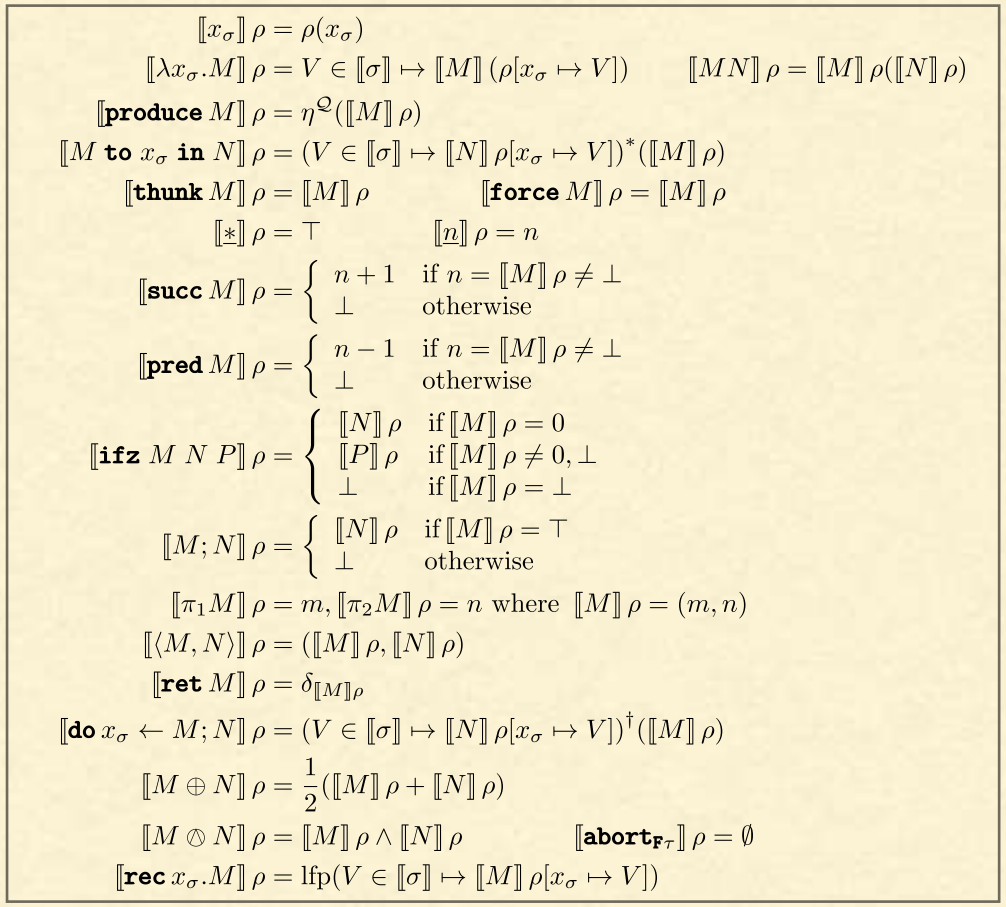
$$
\begin{aligned}
[\![x_\sigma]\!]\,\rho &= \rho(x_\sigma)\\
[\![\lambda x_\sigma.M]\!]\,\rho &= V \in [\![\sigma]\!] \mapsto [\![M]\!]\,(\rho[x_\sigma \mapsto V]) \qquad [\![M N]\!]\,\rho = [\![M]\!]\,\rho([\![N]\!]\,\rho)\\
[\![\mathbf{produce}\, M]\!]\,\rho &= \eta^{\mathcal{Q}}([\![M]\!]\,\rho)\\
[\![M\ \mathbf{to}\ x_\sigma\ \mathbf{in}\ N]\!]\,\rho &= (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^*([\![M]\!]\,\rho)\\
[\![\mathbf{thunk}\, M]\!]\,\rho &= [\![M]\!]\,\rho \qquad [\![\mathbf{force}\, M]\!]\,\rho = [\![M]\!]\,\rho\\
[\![\underline{*}]\!]\,\rho &= \top \qquad [\![\underline{n}]\!]\,\rho = n\\
[\![\mathbf{succ}\, M]\!]\,\rho &= \begin{cases} n+1 & \text{if } n = [\![M]\!]\,\rho \neq \bot\\ \bot & \text{otherwise}\end{cases}\\
[\![\mathbf{pred}\, M]\!]\,\rho &= \begin{cases} n-1 & \text{if } n = [\![M]\!]\,\rho \neq \bot\\ \bot & \text{otherwise}\end{cases}\\
[\![\mathbf{ifz}\ M\ N\ P]\!]\,\rho &= \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = 0\\ [\![P]\!]\,\rho & \text{if } [\![M]\!]\,\rho \neq 0, \bot\\ \bot & \text{if } [\![M]\!]\,\rho = \bot\end{cases}\\
[\![M; N]\!]\,\rho &= \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = \top\\ \bot & \text{otherwise}\end{cases}\\
[\![\pi_1 M]\!]\,\rho &= m, [\![\pi_2 M]\!]\,\rho = n \text{ where } [\![M]\!]\,\rho = (m, n)\\
[\![\langle M, N\rangle]\!]\,\rho &= ([\![M]\!]\,\rho, [\![N]\!]\,\rho)\\
[\![\mathbf{ret}\, M]\!]\,\rho &= \delta_{[\![M]\!]\rho}\\
[\![\mathbf{do}\, x_\sigma \leftarrow M; N]\!]\,\rho &= (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^\dagger([\![M]\!]\,\rho)\\
[\![M \oplus N]\!]\,\rho &= \frac{1}{2}([\![M]\!]\,\rho + [\![N]\!]\,\rho)\\
[\![M \owedge N]\!]\,\rho &= [\![M]\!]\,\rho \wedge [\![N]\!]\,\rho \qquad [\![\mathbf{abort}_{\mathbf{F}\tau}]\!]\,\rho = \emptyset\\
[\![\mathbf{rec}\, x_\sigma.M]\!]\,\rho &= \mathrm{lfp}(V \in [\![\sigma]\!] \mapsto [\![M]\!]\,\rho[x_\sigma \mapsto V])
\end{aligned}
$$

ADEQUACY

- **Prop (adequacy).**
  For every M : **FVunit**,
  — $[\![M]\!]=\bot$ and $\Pr(M\downarrow)=0$, or
  — $[\![M]\!]=\varnothing$ and $\Pr(M\downarrow)=1$, or else
  — $\Pr(M\downarrow)=\min\{\nu(\{\top\}) \mid \nu \in [\![M]\!]\}$

- I.e., $\Pr(M\downarrow)=h^*([\![M]\!])$
  where $h(\nu) = \nu(\{\top\})$

- Proof: by suitable logical relations.



$$[\![x_\sigma]\!]\,\rho = \rho(x_\sigma)$$

$$[\![\lambda x_\sigma.M]\!]\,\rho = V \in [\![\sigma]\!] \mapsto [\![M]\!]\,(\rho[x_\sigma \mapsto V]) \qquad [\![MN]\!]\,\rho = [\![M]\!]\,\rho([\![N]\!]\,\rho)$$

$$[\![\mathtt{produce}\,M]\!]\,\rho = \eta^{\mathcal{Q}}([\![M]\!]\,\rho)$$

$$[\![M\ \mathtt{to}\ x_\sigma\ \mathtt{in}\ N]\!]\,\rho = (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^*([\![M]\!]\,\rho)$$

$$[\![\mathtt{thunk}\,M]\!]\,\rho = [\![M]\!]\,\rho \qquad [\![\mathtt{force}\,M]\!]\,\rho = [\![M]\!]\,\rho$$

$$[\![\underline{*}]\!]\,\rho = \top \qquad [\![\underline{n}]\!]\,\rho = n$$

$$[\![\mathtt{succ}\,M]\!]\,\rho = \begin{cases} n+1 & \text{if } n = [\![M]\!]\,\rho \neq \bot \\ \bot & \text{otherwise} \end{cases}$$

$$[\![\mathtt{pred}\,M]\!]\,\rho = \begin{cases} n-1 & \text{if } n = [\![M]\!]\,\rho \neq \bot \\ \bot & \text{otherwise} \end{cases}$$

$$[\![\mathtt{ifz}\ M\ N\ P]\!]\,\rho = \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = 0 \\ [\![P]\!]\,\rho & \text{if } [\![M]\!]\,\rho \neq 0, \bot \\ \bot & \text{if } [\![M]\!]\,\rho = \bot \end{cases}$$

$$[\![M; N]\!]\,\rho = \begin{cases} [\![N]\!]\,\rho & \text{if } [\![M]\!]\,\rho = \top \\ \bot & \text{otherwise} \end{cases}$$

$$[\![\pi_1 M]\!]\,\rho = m, [\![\pi_2 M]\!]\,\rho = n \text{ where } [\![M]\!]\,\rho = (m, n)$$

$$[\![\langle M, N\rangle]\!]\,\rho = ([\![M]\!]\,\rho, [\![N]\!]\,\rho)$$

$$[\![\mathtt{ret}\,M]\!]\,\rho = \delta_{[\![M]\!]\rho}$$

$$[\![\mathtt{do}\,x_\sigma \leftarrow M; N]\!]\,\rho = (V \in [\![\sigma]\!] \mapsto [\![N]\!]\,\rho[x_\sigma \mapsto V])^\dagger([\![M]\!]\,\rho)$$

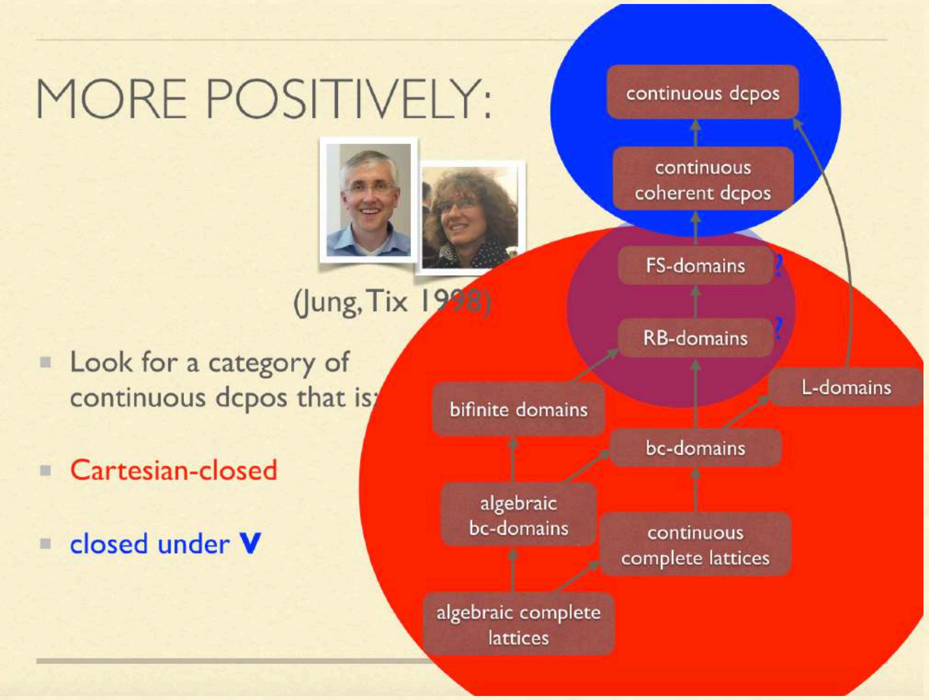
$$[\![M \oplus N]\!]\,\rho = \frac{1}{2}([\![M]\!]\,\rho + [\![N]\!]\,\rho)$$

$$[\![M \owedge N]\!]\,\rho = [\![M]\!]\,\rho \wedge [\![N]\!]\,\rho \qquad [\![\mathtt{abort}_{\mathtt{F}\tau}]\!]\,\rho = \emptyset$$

$$[\![\mathtt{rec}\,x_\sigma.M]\!]\,\rho = \mathrm{lfp}(V \in [\![\sigma]\!] \mapsto [\![M]\!]\,\rho[x_\sigma \mapsto V])$$

NOTE

- None of that yet requires CCCs of **continuous** (or algebraic) domains



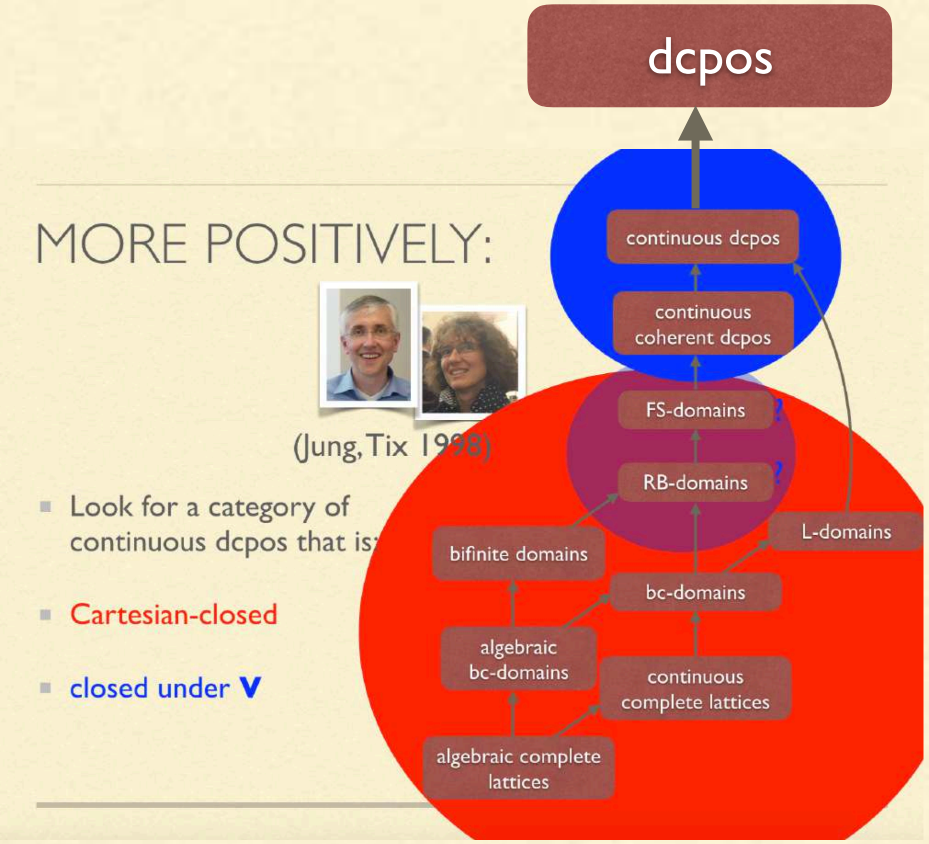
MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under $\vee$

NOTE

- None of that yet requires CCCs of **continuous** (or algebraic) domains
- Soundness/adequacy works even for non-call-by-push-value probabilistic languages, working in the CCC **Dcpo**



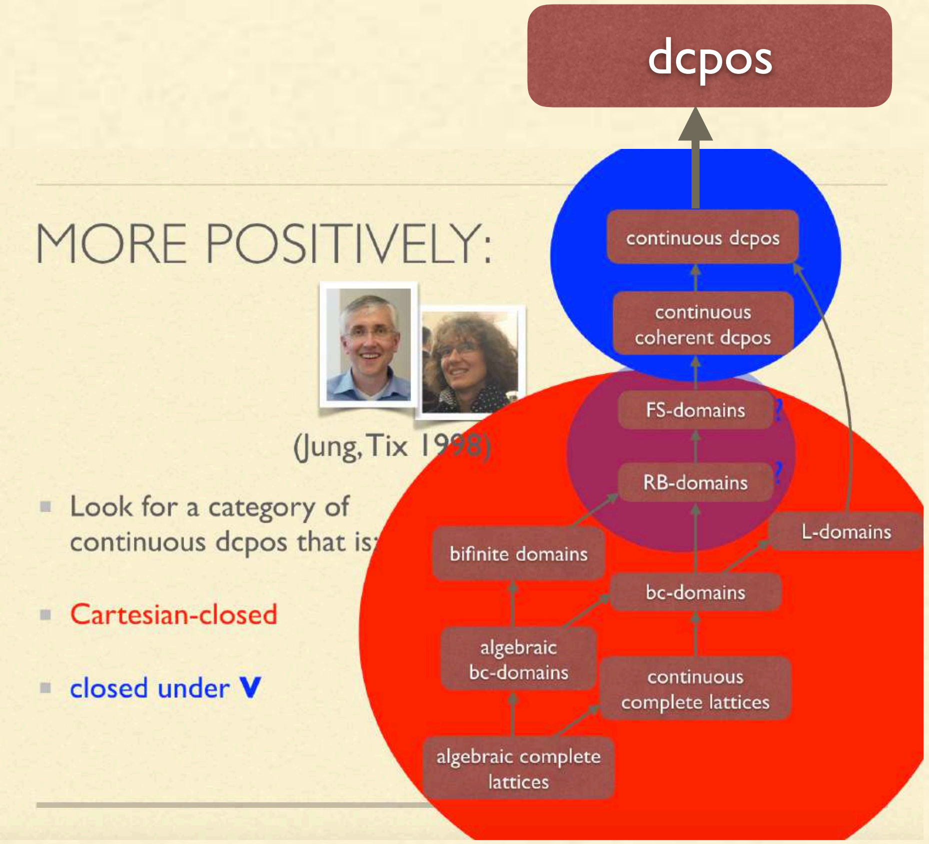
MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **V**

NOTE

- None of that yet requires CCCs of **continuous** (or algebraic) domains
- Soundness/adequacy works even for non-call-by-push-value probabilistic languages, working in the CCC **Dcpo**



MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **V**

- Continuity is only needed for more advanced applications:
— full abstraction (next)
— commutativity of the **V** monad (Fubini) at higher types

THE CONTEXTUAL PREORDER

- Let $M \lesssim N$ iff for every context C of output type **FVunit**,
$$\Pr(C\,.\,M\downarrow) \leq \Pr(C\,.\,N\downarrow)$$

THE CONTEXTUAL PREORDER

- Let $M \leq N$ iff for every context C of output type **FVunit**,

  $$\Pr(C\,.\,M{\downarrow}) \leq \Pr(C\,.\,N{\downarrow})$$

- $M \leq N$ iff for every context C of output type **FVunit**,

  $$h^*([\![C[M]]\!]) \leq h^*([\![C[N]]\!]) \qquad \text{(adequacy)}$$

THE CONTEXTUAL PREORDER

- Let $M \precsim N$ iff for every context C of output type **FVunit**,

  $$\Pr(C \,.\, M{\downarrow}) \leq \Pr(C \,.\, N{\downarrow})$$

- $M \precsim N$ iff for every context C of output type **FVunit**,

  $$h^*(\llbracket C[M] \rrbracket) \leq h^*(\llbracket C[N] \rrbracket) \qquad \text{(adequacy)}$$

- **Corollary.** If $\llbracket M \rrbracket \leq \llbracket N \rrbracket$ then $M \precsim N$.

THE CONTEXTUAL PREORDER

- Let $M \lesssim N$ iff for every context C of output type **FVunit**,
  $$\Pr(C\,.\,M{\downarrow}) \leq \Pr(C\,.\,N{\downarrow})$$

- $M \lesssim N$ iff for every context C of output type **FVunit**,
  $$h^*([\![C[M]]\!]) \leq h^*([\![C[N]]\!]) \qquad \text{(adequacy)}$$

- **Corollary.** If $[\![M]\!] \leq [\![N]\!]$ then $M \lesssim N$.

- Proof. $[\![C[M]]\!] = [\![C]\!]\,([\![M]\!]) \leq [\![C]\!]\,([\![N]\!]) = [\![C[N]]\!]$
  since $[\![C]\!]$ ($= [\![\lambda x\,.\,C[x]]\!]$) is Scott-continuous hence monotonic.
  Then apply h^*, which is monotonic as well. □

FULL ABSTRACTION?

- **Conjecture (full abstraction):** $[\![M]\!] \leq [\![N]\!]$ iff $M \leq N$.

FULL ABSTRACTION?

- **Conjecture (full abstraction):** $[\![M]\!] \leq [\![N]\!]$ iff $M \leq N$.

- **Wrong.**

  — missing parallel if (**pifz**), as in (Plotkin77)

  — even with **pifz**, missing *statistical termination testers* $\bigcirc_{>b}$,
  as in (GL15):

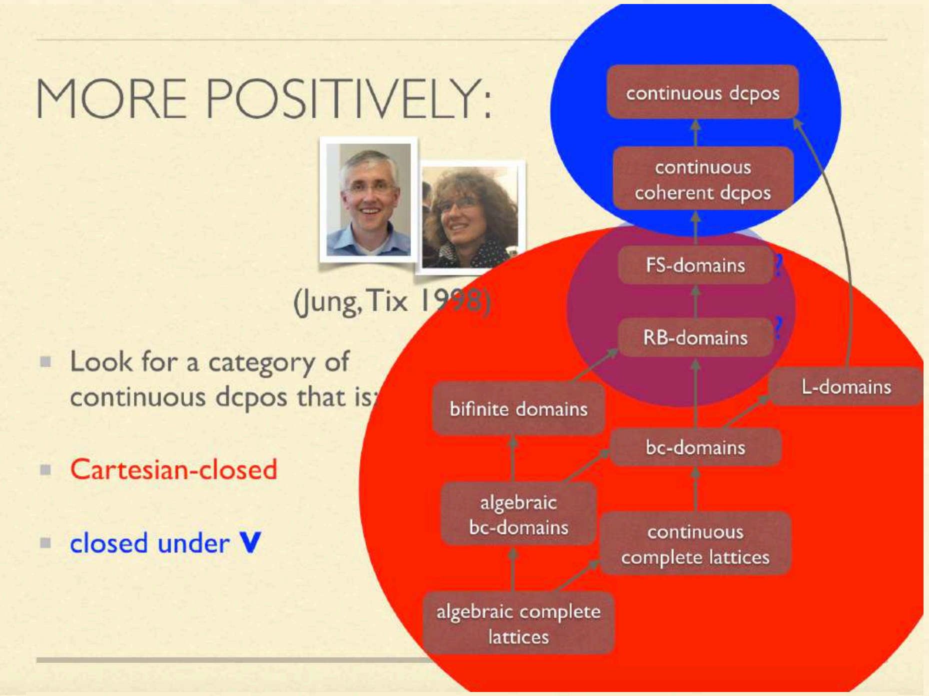
  $\bigcirc_{>b} M$ terminates if M terminates with prob. $>b$,
  otherwise does not terminate.

FULL ABSTRACTION

- Adding **pifz** + $\bigcirc_{>b}$,
- **Theorem (full abstraction):** with **pifz** and $\bigcirc_{>b}$,
  $[\![M]\!] \leq [\![N]\!]$ iff $M \preceq N$.
- For the argument, see the paper.
  Uses the deep structure of continuous coherent dcpos and continuous complete lattices.
  Core: theorems on (effective) coincidence of topologies.

SUMMARY

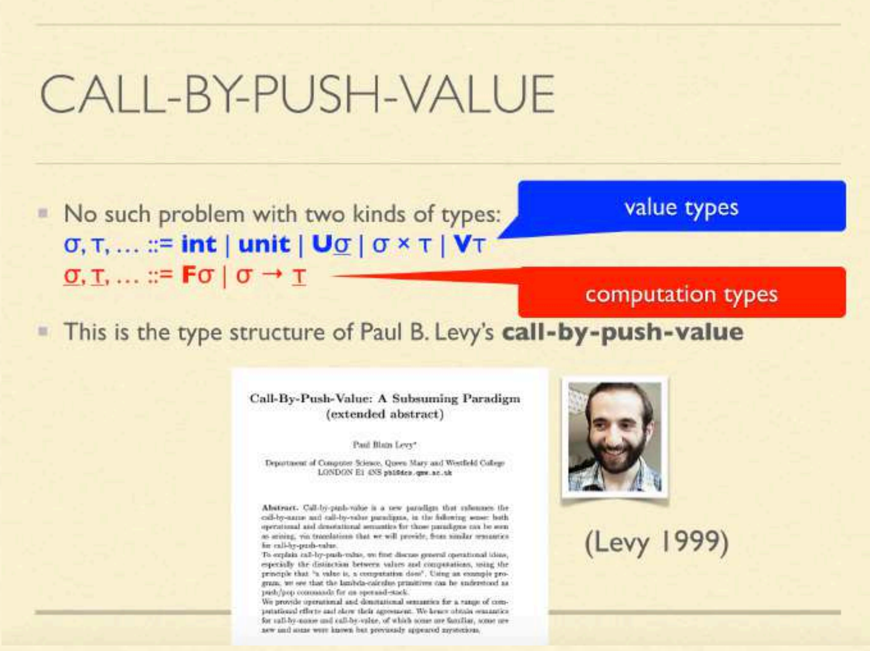
- Circumventing the trouble with **V** by using <u>two</u> classes of types, as provided by call-by-push-value
- We obtain (inequational) full abstraction with prob. choice + demonic non-determinism
- Questions?



MORE POSITIVELY:

(Jung, Tix 1998)

- Look for a category of continuous dcpos that is:
- Cartesian-closed
- closed under **V**



CALL-BY-PUSH-VALUE

- No such problem with two kinds of types:
  $\sigma, \tau, \ldots ::= \mathbf{int} \mid \mathbf{unit} \mid \mathbf{U}\underline{\sigma} \mid \sigma \times \tau \mid \mathbf{V}\tau$
  $\underline{\sigma}, \underline{\tau}, \ldots ::= \mathbf{F}\sigma \mid \sigma \to \underline{\tau}$
- This is the type structure of Paul B. Levy's **call-by-push-value**

Call-By-Push-Value: A Subsuming Paradigm
(extended abstract)

Paul Blain Levy*

Department of Computer Science, Queen Mary and Westfield College
LONDON E1 4NS

(Levy 1999)