UNIVERSITY OF Southampton
School of Electronics and Computer Science

THE WIRE CALCULUS

Paweł Sobociński

11/09/09, IFIP 1.3 Meeting, Udine

PROCESS CALCULI

- CCS 80s, Pi 90s, ambients, ...
- **common features**:
  - syntactic expressions represent “processes”
    - dynamics a first class entity (eg prefix & + in CCS)
  - (structural) operational semantics
    - observational equivalence, (weak) bisimulation
    - (weak) bisimulation induces an algebra on syntactic terms
  - Dijkstra-Hoare-Milner parallel composition ||

ALGEBRA OF PROCESSES

- Milner's SOS:

$$\frac{P \xrightarrow{a} P'}{P \| Q \xrightarrow{a} P' \| Q} \ (\|\mathrm{L}) \qquad \frac{Q \xrightarrow{a} Q'}{P \| Q \xrightarrow{a} P \| Q'} \ (\|\mathrm{R}) \qquad \frac{P \xrightarrow{a} P' \quad Q \xrightarrow{\overline{a}} Q'}{P \| Q \xrightarrow{\tau} P' \| Q'} \ (\textsc{Par})$$

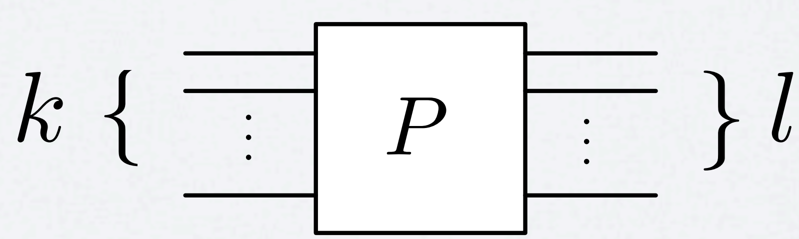
- what is the algebra induces by (weak) bisimilarity?
- the operation || is a commutative monoid
  - ie. processes live in a "chemical soup"
  - is it always the case that **concurrent universe = chemical soup**?
- interleaving gives **concurrency = nondeterminism**, always reasonable?
  - implementation issues: || is a very powerful scheduler

ANOTHER NOTION OF PROCESS

- Let **Σ** be a set of *signals* + silent action ι for int. computation
- for $k, l \in \mathbb{N}$ a (k, l)-transition is a labelled transition of the form

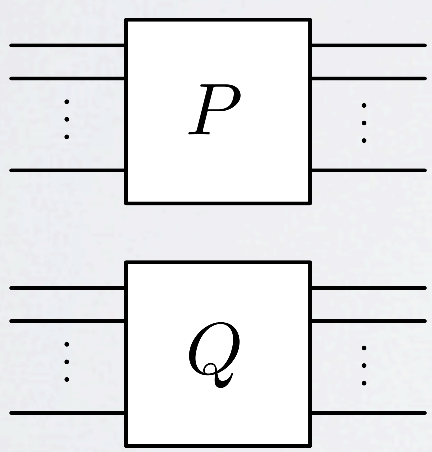
$$P \xrightarrow[\vec{b}]{\vec{a}} Q, \quad \#(\vec{a}) = k, \#(\vec{b}) = l$$

- any process is the wire calculus has a sort (k, l) and its semantics will be an LTS of (k, l)-transitions



ANOTHER PARALLEL COMPOSITION

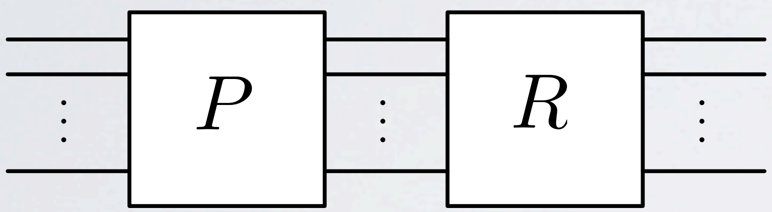
- processes are boxes with left and right boundary
- operators of the calculus allow us to specify & connect boxes
- labels have a monoidal structure (juxtaposition)



$$\frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \qquad R \xrightarrow[\mathbf{d}]{\mathbf{c}} S}{P \otimes R \xrightarrow[\mathbf{bd}]{\mathbf{ac}} Q \otimes S} \ (\textsc{Ten})$$

- $\otimes$ neither commutative nor interleaving

SYNCHRONISING ON BOUNDARY



$$\frac{P \xrightarrow[\mathbf{c}]{\mathbf{a}} Q \qquad R \xrightarrow[\mathbf{b}]{\mathbf{c}} S}{P;R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q;S} \ (\textsc{Cut})$$

- non commutative
- don't confuse with seq composition in imperative prog langs

DYNAMICS - CHOICE

- CSP-like (□) external choice

$$\frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \qquad (\mathbf{ab} \neq \iota)}{P+R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q} \ (+\mathrm{L}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \qquad (\mathbf{ab} \neq \iota)}{R+P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q} \ (+\mathrm{R})$$

$$\frac{P \xrightarrow[\iota]{\iota} Q \qquad R \xrightarrow[\iota]{\iota} S}{P+R \xrightarrow[\iota]{\iota} Q+S} \ (+\iota)$$

DYNAMICS - PREFIX

- signals live in some set Σ
- prefix strings: $M \ ::= \ \epsilon \mid x \mid \lambda x \mid \iota \mid \sigma \in \Sigma \mid MM$
- prefixes: $P \ ::= \ \ldots \mid \frac{M}{M}P$
- $\frac{u}{v}P$ with λx in u or v binds free occurrences of x in P
- substitution: $\sigma : bd(\frac{u}{v}) \to \Sigma + \{\iota\}$

$$\frac{}{\frac{u}{v}P \xrightarrow[v|_\sigma]{u|_\sigma} P|_\sigma} \ (\textsc{Pref})$$

Example: $\frac{\lambda x a}{\lambda y}P \ \xrightarrow[\beta]{\alpha\, a} \ P[\alpha/x, \beta/y]$ for all $\alpha, \beta \in \Sigma$

DYNAMICS - RECURSION

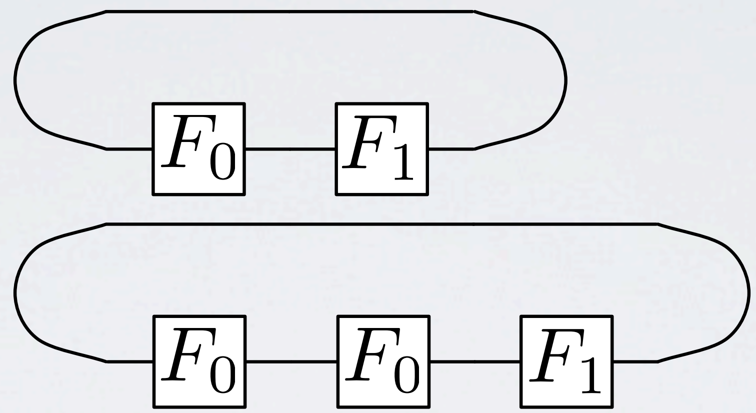
$$\frac{P[\mu Y.\, P/Y] \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}{\mu Y.\, P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\textsc{Rec})$$

- Standard Plotkin SOS rule

EXAMPLE - BASIC WIRES

Picture	Expression	Behaviour
	$I \stackrel{\text{def}}{=} \mu Y.\ \frac{\lambda x}{\lambda x} Y$	$\dfrac{}{I \xrightarrow[a]{a} I}$ (ID)
	$\mathsf{d} \stackrel{\text{def}}{=} \mu Y.\ \frac{}{\lambda x \lambda x} Y$	$\dfrac{}{\mathsf{d} \xrightarrow[a\,a]{} \mathsf{d}}$ (d)
	$\mathsf{e} \stackrel{\text{def}}{=} \mu Y.\ \frac{\lambda x \lambda x}{} Y$	$\dfrac{}{\mathsf{e} \xrightarrow{a\,a} \mathsf{e}}$ (e)

CONCURRENCY RULES



$$\frac{}{P \xrightarrow[\iota]{\iota} P}\ (\textsc{Refl})$$

unconnected processes cannot block each other

$$\frac{P \xrightarrow[\iota]{\iota} R \quad R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\iota\mathrm{L}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} R \quad R \xrightarrow[\iota]{\iota} Q}{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\iota\mathrm{R})$$

processes are not assumed to run at the same speed

SUMMARY - THE WIRE CALCULUS

$$P \ ::= \ Y \mid P \,;\, P \mid P \otimes P \mid \tfrac{M}{M} P \mid P + P \mid \mu Y : \tau.\, P$$
$$M \ ::= \ \epsilon \mid x \mid \lambda x \mid \iota \mid \sigma \in \Sigma$$

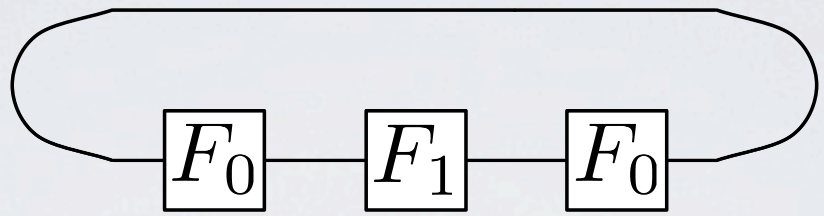
$$\frac{}{P \xrightarrow[\iota]{\iota} P}\ (\textsc{Refl}) \qquad \frac{P \xrightarrow[\iota]{\iota} R \quad R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\iota\mathrm{L}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} R \quad R \xrightarrow[\iota]{\iota} Q}{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\iota\mathrm{R})$$

$$\frac{P \xrightarrow[\mathbf{c}]{\mathbf{a}} Q \quad R \xrightarrow[\mathbf{b}]{\mathbf{c}} S}{P;R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q;S}\ (\textsc{Cut}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \quad R \xrightarrow[\mathbf{d}]{\mathbf{c}} S}{P \otimes R \xrightarrow[\mathbf{bd}]{\mathbf{ac}} Q \otimes S}\ (\textsc{Ten})$$

$$\frac{}{\frac{u}{v} P \xrightarrow[v|\sigma]{u|\sigma} P|_\sigma}\ (\textsc{Pref}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \quad (\mathbf{ab} \neq \iota)}{P + R \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (+\mathrm{L}) \qquad \frac{P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q \quad (\mathbf{ab} \neq \iota)}{R + P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (+\mathrm{R})$$

$$\frac{P \xrightarrow[\iota]{\iota} Q \quad R \xrightarrow[\iota]{\iota} S}{P + R \xrightarrow[\iota]{\iota} Q + S}\ (+\iota) \qquad \frac{P[\mu Y.\, P/Y] \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}{\mu Y.\, P \xrightarrow[\mathbf{b}]{\mathbf{a}} Q}\ (\textsc{Rec})$$

EXAMPLE



$$\frac{}{F_0 \xrightarrow[0]{0} F_0}\ (0\textsc{set}0) \qquad \frac{}{F_0 \xrightarrow[0]{1} F_1}\ (0\textsc{set}1) \qquad \frac{}{F_0 \xrightarrow[\iota]{\iota} F_0}\ (0\textsc{Refl})$$

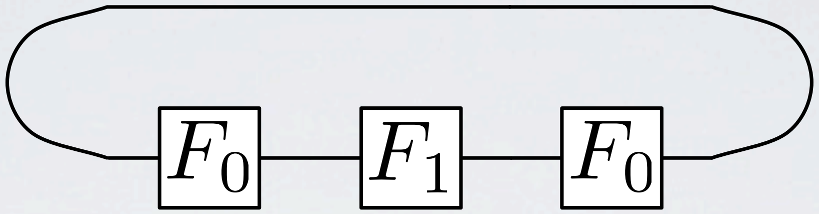
$$\frac{}{F_1 \xrightarrow[1]{1} F_1}\ (1\textsc{set}1) \qquad \frac{}{F_1 \xrightarrow[1]{0} F_0}\ (1\textsc{set}0) \qquad \frac{}{F_1 \xrightarrow[\iota]{\iota} F_1}\ (1\textsc{Refl})$$

Expressions

$$F_0 \stackrel{\text{def}}{=} \mu Y.\, \tfrac{0}{0}Y + \tfrac{1}{0}\mu Z.\, (\tfrac{1}{1}Z + \tfrac{0}{1}Y) \qquad F_1 \stackrel{\text{def}}{=} \mu Z.\, \tfrac{1}{1}Z + \tfrac{0}{1}\mu Y.\, (\tfrac{0}{0}Y + \tfrac{1}{0}Z)$$

$$\mathsf{d} \;;\; I \otimes (F_0 \;;\; F_1 \;;\; F_0) \;;\; \mathsf{e}$$

EXAMPLE CTD



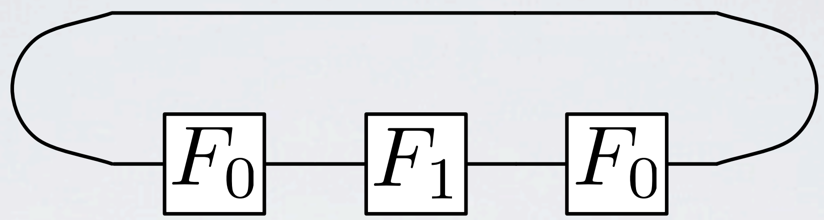
$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut})$$

$$y = 1$$

$$z = 0$$

$$Y = F_0$$

EXAMPLE CTD

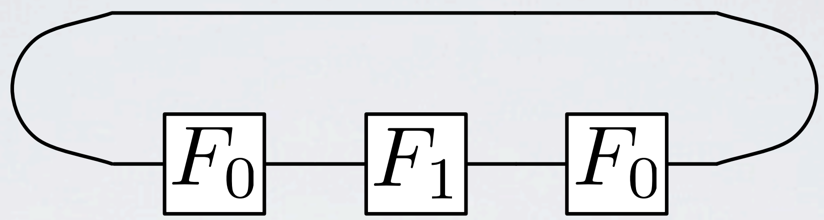


$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut})$$

$$y = 1$$
$$z = 0$$
$$Y = F_0$$

$$\frac{F_0 \xrightarrow[0]{x} X \qquad F_1 \xrightarrow[1]{0} F_0}{F_0;F_1 \xrightarrow[1]{x} X;F_0}\ (\textsc{Cut})$$

EXAMPLE CTD



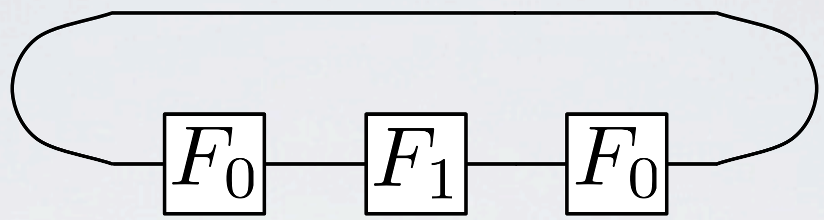
$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut})$$

$$y = 1$$
$$z = 0$$
$$Y = F_0$$

$$\frac{F_0 \xrightarrow[0]{x} X \qquad F_1 \xrightarrow[1]{0} F_0}{F_0;F_1 \xrightarrow[1]{x} X;F_0}\ (\textsc{Cut})$$

$$\frac{F_0;F_1 \xrightarrow[1]{x} X;F_0 \qquad F_0 \xrightarrow[0]{1} F_1}{F_0;F_1;F_0 \xrightarrow[0]{x} X;F_0;F_1}\ (\textsc{Cut})$$

EXAMPLE CTD



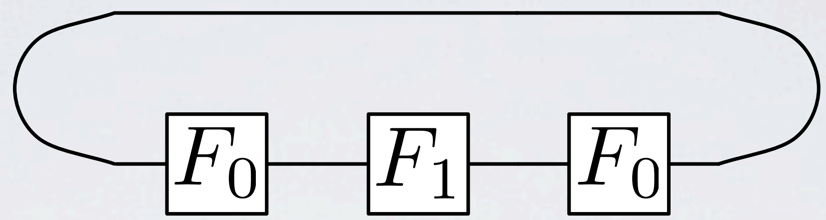
$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut}) \qquad \begin{array}{c} y = 1 \\ z = 0 \\ Y = F_0 \end{array}$$

$$\frac{F_0 \xrightarrow[0]{x} X \qquad F_1 \xrightarrow[1]{0} F_0}{F_0;F_1 \xrightarrow[1]{x} X;F_0}\ (\textsc{Cut})$$

$$\frac{F_0;F_1 \xrightarrow[1]{x} X;F_0 \qquad F_0 \xrightarrow[0]{1} F_1}{F_0;F_1;F_0 \xrightarrow[0]{x} X;F_0;F_1}\ (\textsc{Cut})$$

$$\frac{}{I \otimes (F_0;F_1;F_0) \xrightarrow[w0]{wx} I \otimes (X;F_0;F_1)}\ (\otimes)$$

EXAMPLE CTD



$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut}) \qquad \begin{array}{c} y = 1 \\ z = 0 \\ Y = F_0 \end{array}$$

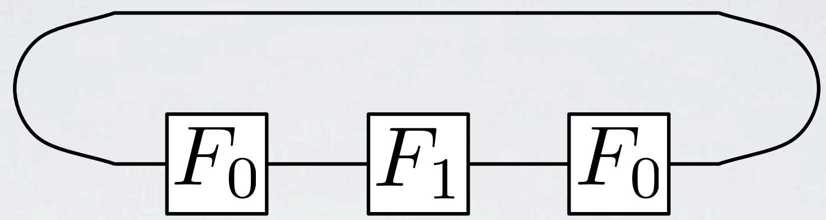
$$\frac{F_0 \xrightarrow[0]{x} X \qquad F_1 \xrightarrow[1]{0} F_0}{F_0;F_1 \xrightarrow[1]{x} X;F_0}\ (\textsc{Cut})$$

$$\frac{F_0;F_1 \xrightarrow[1]{x} X;F_0 \qquad F_0 \xrightarrow[0]{1} F_1}{F_0;F_1;F_0 \xrightarrow[0]{x} X;F_0;F_1}\ (\textsc{Cut})$$

$$\frac{}{I\otimes(F_0;F_1;F_0) \xrightarrow[w0]{wx} I\otimes(X;F_0;F_1)}\ (\otimes)$$

$$\frac{}{(I\otimes(F_0;F_1;F_0));\mathsf{e} \xrightarrow{0x} I\otimes(X;F_0;F_1);\mathsf{e}}\ (\textsc{Cut})$$

EXAMPLE CTD



$$\frac{F_0 \xrightarrow[z]{x} X \qquad F_1 \xrightarrow[y]{z} Y}{F_0;F_1 \xrightarrow[y]{x} X;Y}\ (\textsc{Cut}) \qquad \begin{array}{c} y = 1 \\ z = 0 \\ Y = F_0 \end{array}$$

$$\frac{F_0 \xrightarrow[0]{x} X \qquad F_1 \xrightarrow[1]{0} F_0}{F_0;F_1 \xrightarrow[1]{x} X;F_0}\ (\textsc{Cut})$$

$$\frac{F_0;F_1 \xrightarrow[1]{x} X;F_0 \qquad F_0 \xrightarrow[0]{1} F_1}{F_0;F_1;F_0 \xrightarrow[0]{x} X;F_0;F_1}\ (\textsc{Cut})$$

$$\frac{}{I\otimes(F_0;F_1;F_0) \xrightarrow[w0]{wx} I\otimes(X;F_0;F_1)}\ (\otimes)$$

$$\frac{}{(I\otimes(F_0;F_1;F_0));\mathsf{e} \xrightarrow{0x} I\otimes(X;F_0;F_1);\mathsf{e}}\ (\textsc{Cut})$$

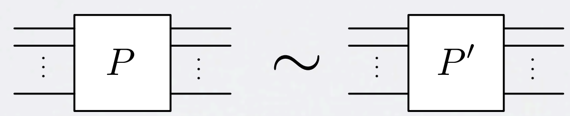
$$\frac{}{\mathsf{d};(I\otimes(F_0;F_1;F_0));\mathsf{e} \longrightarrow \mathsf{d};I\otimes(F_0;F_0;F_1);\mathsf{e}}\ (\textsc{Cut})$$

BISIMILARITY

- bisimilarity = weak bisimilarity
- bisimilarity a congruence wrt all operators of the language

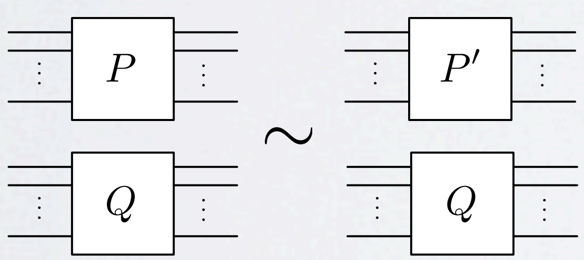
$$P, P' : (k, l) \quad Q : (m, n) \quad R : (l, l') \quad S : (k', k)$$

$$P \sim P'$$

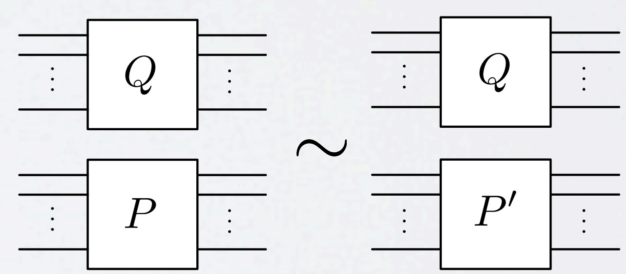


then

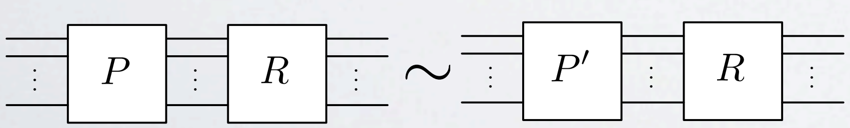
$$P \otimes Q \ \sim \ P' \otimes Q$$



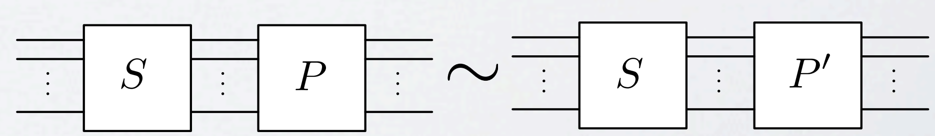
$$Q \otimes P \ \sim \ Q \otimes P'$$



$$P \,;\, R \ \sim \ P' \,;\, R$$



$$S \,;\, P \ \sim \ S \,;\, P'$$



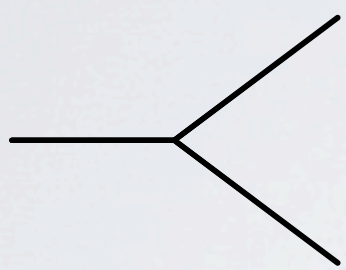
ALGEBRA

- what is the resulting algebra?
  - objects = natural numbers; arrows = equivalence classes wrt bisimilarity
  - a strictly associative monoidal category
  - d and e yield compact closed structure
- easy to define a directed variant of calculus
- See "*A non-interleaving process calculus for multi-party synchronisation*" in Proc. ICE '09

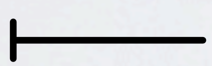
I'VE SEEN THIS ALL BEFORE!

- **related work** (incomplete!):
  - RFC Walters et al:
    - Span(**Graph**) - similar algebra, no SOS, dynamics not first class members of language
  - Gadducci Montanari et al:
    - tiles - similar algebra, same SOS for tensor and composition, dynamics not first class, less structure (more general) wrt weak issues
  - Abramsky et al:
    - interaction categories - similar SOS for tensor and composition, much more involved type structure
  - Arbab
    - Reo, etc: similar modelling style but has semantic issues, "user-defined" dynamics
  - Stefanescu
    - network algebra - monolithic, hard to tell what is primitive

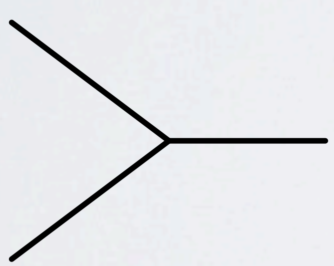
OTHER WIRES



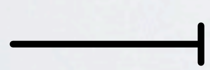
$$\Delta \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x}{\lambda x \lambda x} Y \qquad \frac{}{\Delta \xrightarrow[a\,a]{a} \Delta}\,(\Delta)$$



$$\top \stackrel{\text{def}}{=} \mu Y.\, \tfrac{}{\lambda x} Y \qquad \frac{}{\top \xrightarrow[a]{} \top}\,(\top)$$



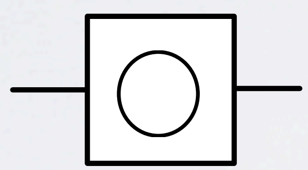
$$\nabla \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x \lambda x}{\lambda x} Y \qquad \frac{}{\nabla \xrightarrow[a]{a\,a} \nabla}\,(\nabla)$$



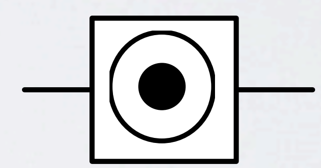
$$\bot \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x}{} Y \qquad \frac{}{\bot \xrightarrow{a} \bot}\,(\bot)$$

PETRI NET PLACES

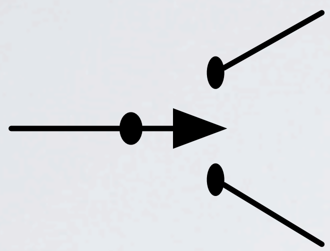
$$\bigcirc \stackrel{\text{def}}{=} \mu Y.\, \frac{\bullet}{\iota}\frac{\iota}{\bullet} Y$$



$$\odot \stackrel{\text{def}}{=} \frac{\iota}{\bullet}\bigcirc$$

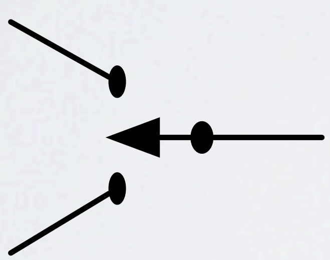


OTHER WIRES



$$\wedge \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x}{\lambda x \iota} Y + \tfrac{\lambda x}{\iota \lambda x} Y \qquad \frac{}{\wedge \xrightarrow[a\iota]{a} \wedge}\;(\wedge \mathrm{L})$$

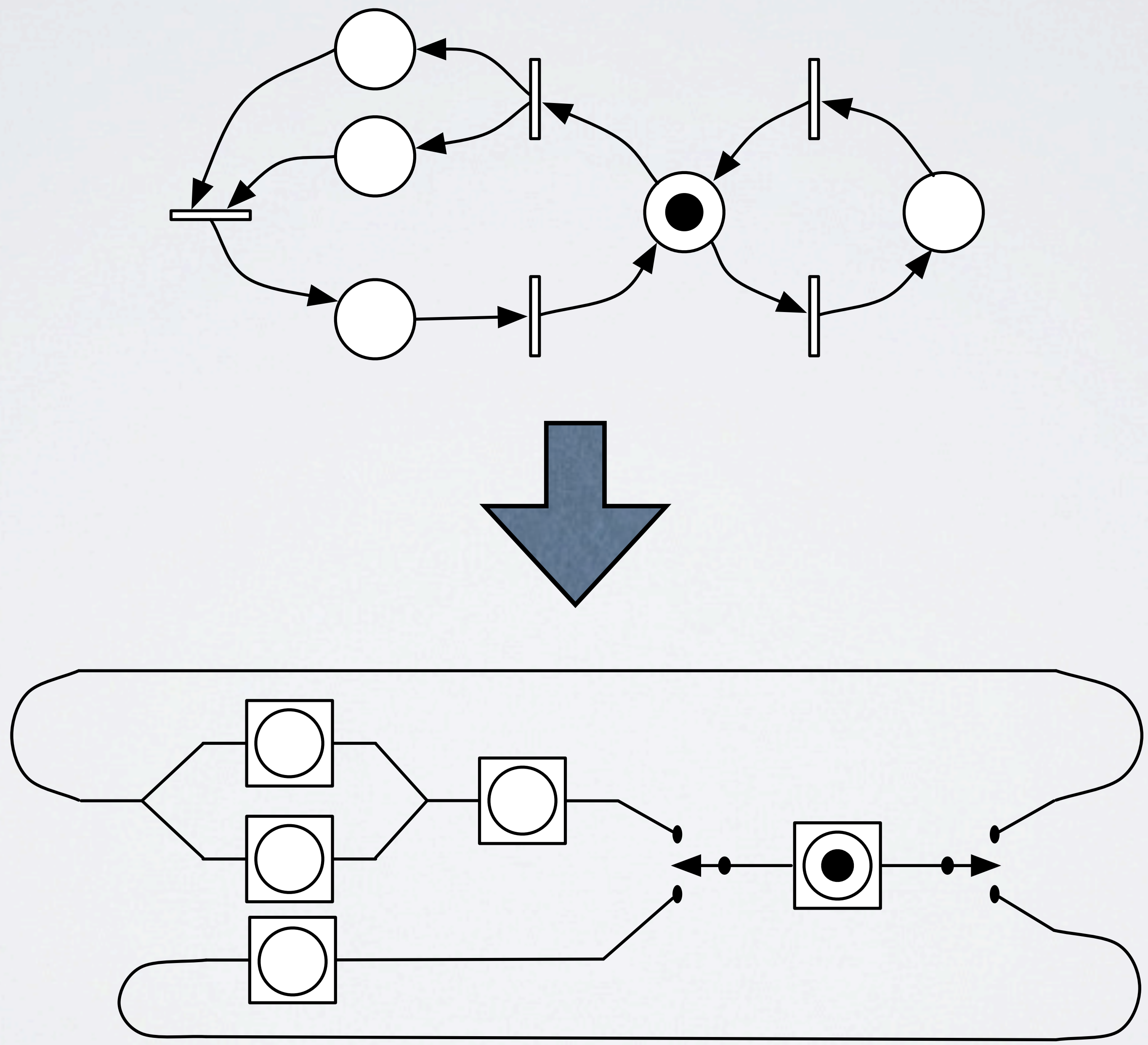
$$\uparrow \stackrel{\text{def}}{=} \mu Y : (0,\, 1).\, Y \qquad \frac{}{\uparrow \xrightarrow[\iota]{} \uparrow}\;(\uparrow)$$



$$\vee \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x \iota}{\lambda x} Y + \tfrac{\iota \lambda x}{\lambda x} Y \qquad \frac{}{\vee \xrightarrow[a]{a\iota} \vee}\;(\vee \mathrm{L})$$

$$\downarrow \stackrel{\text{def}}{=} \mu Y : (1,\, 0).\, Y \qquad \frac{}{\downarrow \xrightarrow{\iota} \downarrow}\;(\downarrow)$$

TRANSLATING NETS



FULL ABSTRACTION

$$M \xrightarrow{e} M' \quad \Longrightarrow \quad [\![M]\!] \rightarrow [\![M']\!]$$

$$[\![M]\!] \rightarrow P$$

$$\Downarrow$$

$$\exists M'.\ [\![M']\!] = P \ \&\ (M' = M \ \vee\ \exists e_1, \ldots, e_m.\ \xrightarrow{e_1} \cdots \xrightarrow{e_m} M')$$

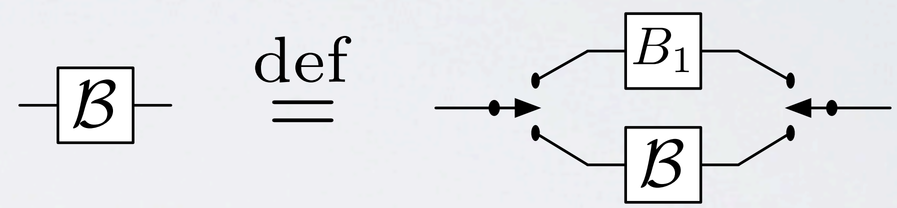
BUFFERS & QUEUES

(inspired by Selinger's Axioms for Asynchrony)

Unbounded buffer

$$B_1 \stackrel{\text{def}}{=} \tfrac{\lambda x}{\iota}\tfrac{\iota}{\lambda x}0$$

$$\mathcal{B} \stackrel{\text{def}}{=} \mu Y.\, B_1 \mid Y$$



$$\frac{}{C[X] \xrightarrow[\iota]{\iota} C[X]} \qquad \frac{\sigma \in \Sigma}{C[X] \xrightarrow[\iota]{\sigma} C[X+\sigma]} \qquad \frac{}{C[X+\sigma] \xrightarrow[\sigma]{\iota} C[X]}$$

Queue

$$\mathcal{Q} \stackrel{\text{def}}{=} \mu Y.\, \tfrac{\lambda x}{\iota}(Y \ ; \ \tfrac{\iota}{\lambda x} I)$$

CONCLUSION

- Wire-calculus: a process calculus that is fundamentally different from existing calculi yet shares many of their features
- mathematically interesting algebra

- future work
  - expressivity
  - continue Selinger's work on axioms for asynchrony