

Process Algrbras

Algebraic semantics to capture the meaning of concurrent systems

- CCS
- pi-calculus
- CSP

Alternatives (many)

- Petri-nets

Calculus of Communicating Systems (CCS)

$\mathcal{N} = \{a, b, c, d, \dots\}$ set of names

$\overline{\mathcal{N}} = \{\bar{a} \mid a \in \mathcal{N}\}$ disjoint set of co-names

$Act = \mathcal{N} \cup \overline{\mathcal{N}} \cup \{\tau\}$ actions

processes

$P ::= 0 \mid \mu.P \mid P + P \mid P \parallel P \mid (\nu a)(P)$

where $\mu \in Act, a \in \mathcal{N}$

Structural Operational Semantics

Smallest relation that satisfies the following rules

$$\frac{}{\mu.P \xrightarrow{\mu} P}$$

$$\frac{P_1 \xrightarrow{\alpha} P'_1}{P_1 + P_2 \xrightarrow{\alpha} P'_1}$$

$$\frac{P_2 \xrightarrow{\alpha} P'_2}{P_1 + P_2 \xrightarrow{\alpha} P'_2}$$

$$\frac{P_1 \xrightarrow{\alpha} P'_1}{P_1 \parallel P_2 \xrightarrow{\alpha} P'_1 \parallel P_2}$$

$$\frac{P_2 \xrightarrow{\alpha} P'_2}{P_1 \parallel P_2 \xrightarrow{\alpha} P_1 \parallel P'_2}$$

$$\frac{P_1 \xrightarrow{a} P'_1 \wedge P_2 \xrightarrow{\bar{a}} P'_2}{P_1 \parallel P_2 \xrightarrow{\alpha} P'_1 \parallel P'_2}$$

$$\frac{P \xrightarrow{\alpha} P' \wedge a \notin \{\alpha, \bar{\alpha}\}}{(\nu a)P \xrightarrow{\alpha} (\nu a)P'}$$

CCS examples

$a.b.0 \parallel \tau.\bar{a}.0 \parallel \bar{b}.\tau.0$



$a.b.0 \parallel \bar{a}.0 \parallel \bar{b}.\tau.0$



$b.0 \parallel 0 \parallel \bar{b}.\tau.0$



$0 \parallel 0 \parallel \tau.0$



$0 \parallel 0 \parallel 0$

$\tau.a.b.0 \parallel \tau.\bar{a}.0 \parallel \bar{b}.\tau.0$



?

CCS examples

$$a.b.0 \parallel \bar{a}.0 \parallel \bar{a}.\bar{b}.0$$

$$(a.0 + b.0) \parallel \bar{a}.0$$

$$(a.0 + b.0) \parallel (\bar{a}.0 + \bar{b}.0)$$

$$(a.b.0 \parallel \bar{a}.0) \parallel \bar{a}.\bar{b}.0$$

$$(\nu a)(a.b.0 \parallel \bar{a}.0) \parallel \bar{a}.\bar{b}.0$$

CCS examples

$$\tau.(a.0 + b.0) \parallel (\bar{a}.0 + \bar{b}.0)$$

$$(\tau.a.0 + \tau.b.0) \parallel (\bar{a}.0 + \bar{b}.0)$$

$$(\tau.a.0 + \tau.b.0) \parallel (\tau.\bar{a}.0 + \tau.\bar{b}.0)$$

CCS examples

$$a.b.0 + b.a.0$$

$$a.0 || b.0$$

$$a.b.c.0 + c.a.b.0$$

$$a.b.0 || c.0$$

Replication

processes

$$P ::= 0 \mid \mu. P \mid P + P \mid P \parallel P \mid (\nu a)(P) \mid !P$$

~~$$\frac{P \xrightarrow{\alpha} P'}{!P \xrightarrow{\alpha} P' \parallel !P}$$~~

~~$$\frac{}{!P \xrightarrow{\tau} P \parallel !P}$$~~

$$\frac{P \parallel !P \xrightarrow{\alpha} P'}{!P \xrightarrow{\alpha} P'}$$

CCS alternative semantics

Here $\equiv$ is a structural congruence (i.e. preserved by syntactic constructs) that satisfy the following actions

$$P \parallel Q \equiv Q \parallel P$$

$$P + Q \equiv Q + P$$

$$P \parallel (Q \parallel R) \equiv (P \parallel Q) \parallel R$$

$$P + (Q + R) \equiv (P + Q) + R$$

$$\frac{P \equiv Q \xrightarrow{\alpha} Q' \equiv P'}{P \xrightarrow{\alpha} P'}$$

$$\frac{}{\mu.P \xrightarrow{\mu} P}$$

$$\frac{P_1 \xrightarrow{\alpha} P'_1}{P_1 \parallel P_2 \xrightarrow{\alpha} P'_1 \parallel P_2}$$

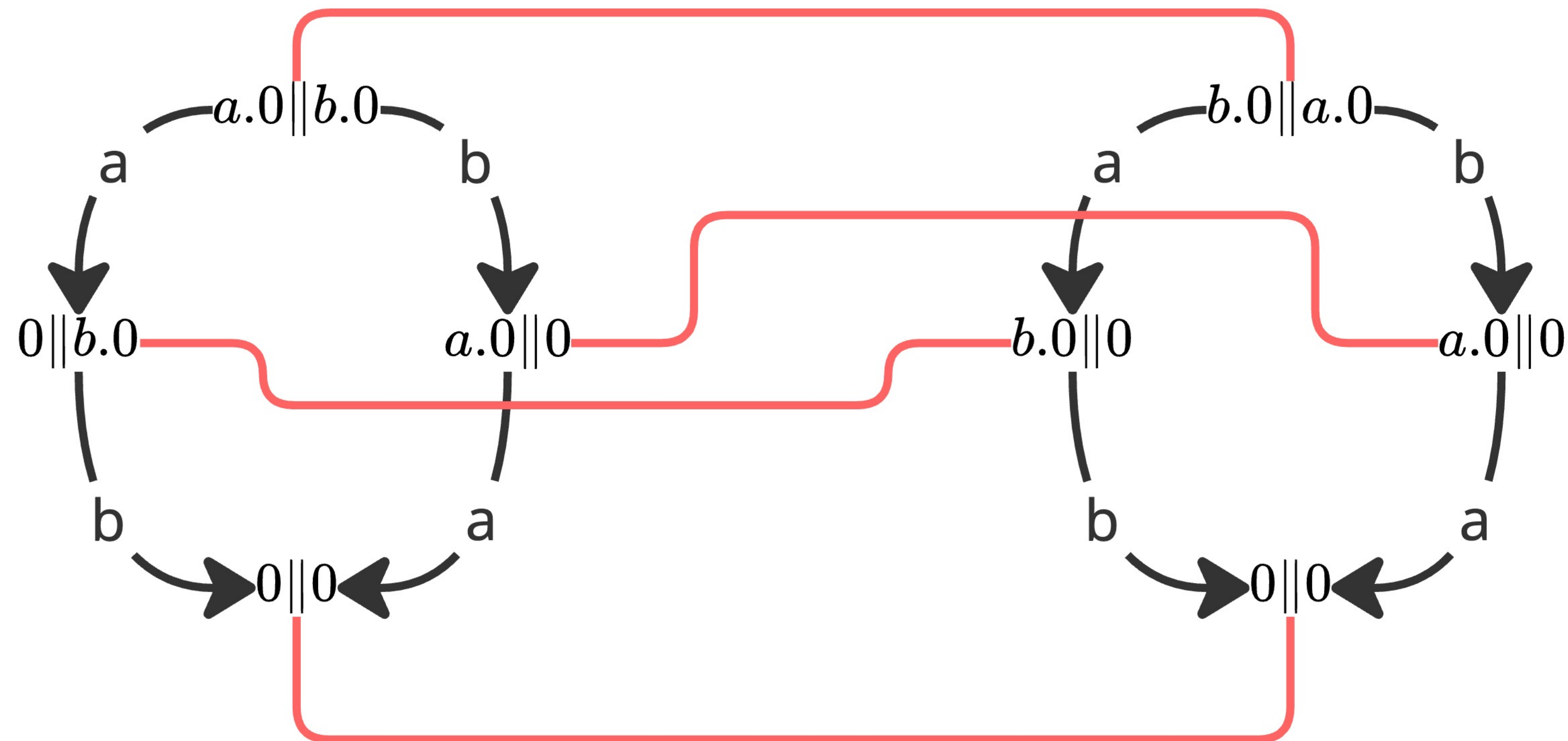
$$\frac{P_1 \xrightarrow{a} P'_1 \wedge P_2 \xrightarrow{\bar{a}} P'_2}{P_1 \parallel P_2 \xrightarrow{\alpha} P'_1 \parallel P'_2}$$

$$\frac{P_1 \xrightarrow{\alpha} P'_1}{P_1 + P_2 \xrightarrow{\alpha} P'_1}$$

$$\frac{P \xrightarrow{\alpha} P' \wedge a \notin \{\alpha, \bar{\alpha}\}}{(\nu a)P \xrightarrow{\alpha} (\nu a)P'}$$

LTS Isomorphism

I.e. there is a label preserving graph-isomorphism between the LTSs



LTS Isomorphism

I.e. there is a label preserving graph-isomorphism between the LTSs

$$a.b.0 + b.a.0$$

$$a.0 || b.0$$

$$a.0 + 0$$

$$a.0$$

$$a.0 || b.0$$

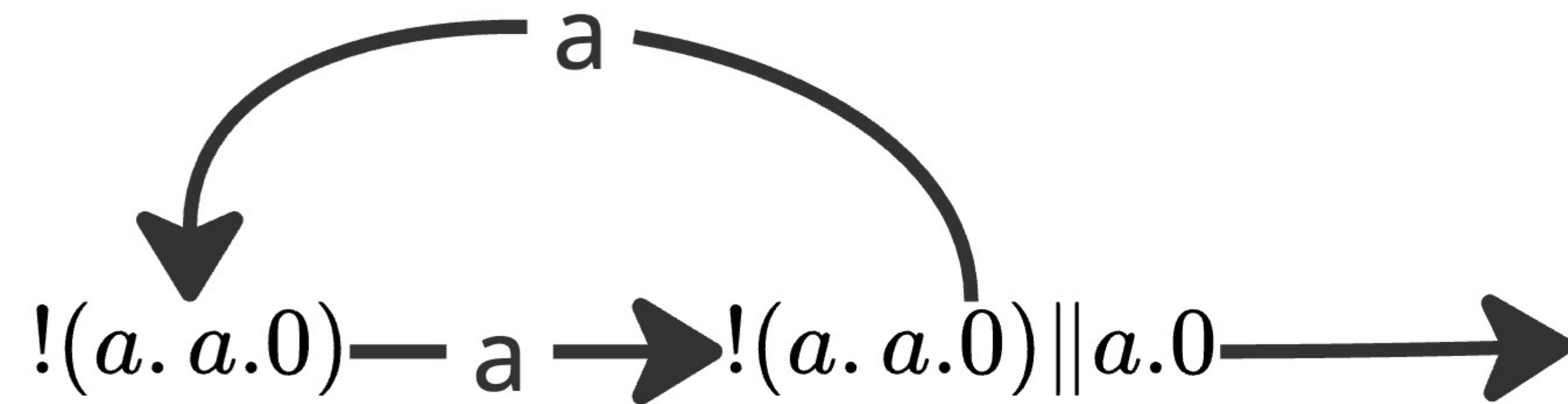
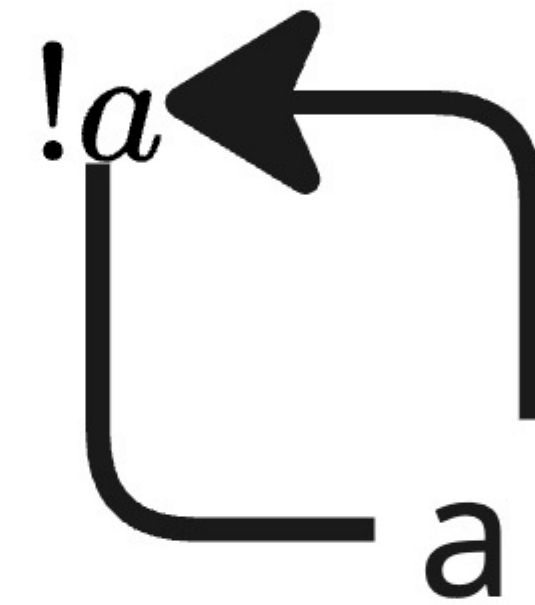
$$b.0 || a.0$$

$$a.0 || 0$$

$$a.0$$

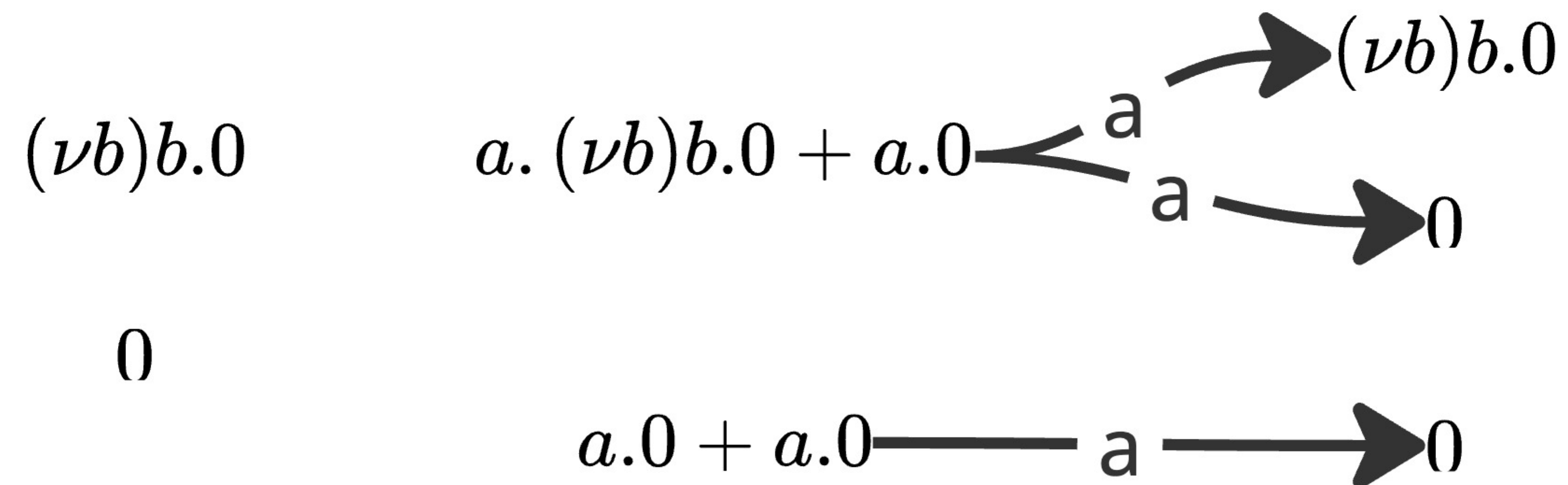
Semantic equivalence

LTS isomorphism may be too restrictive



Semantic equivalence

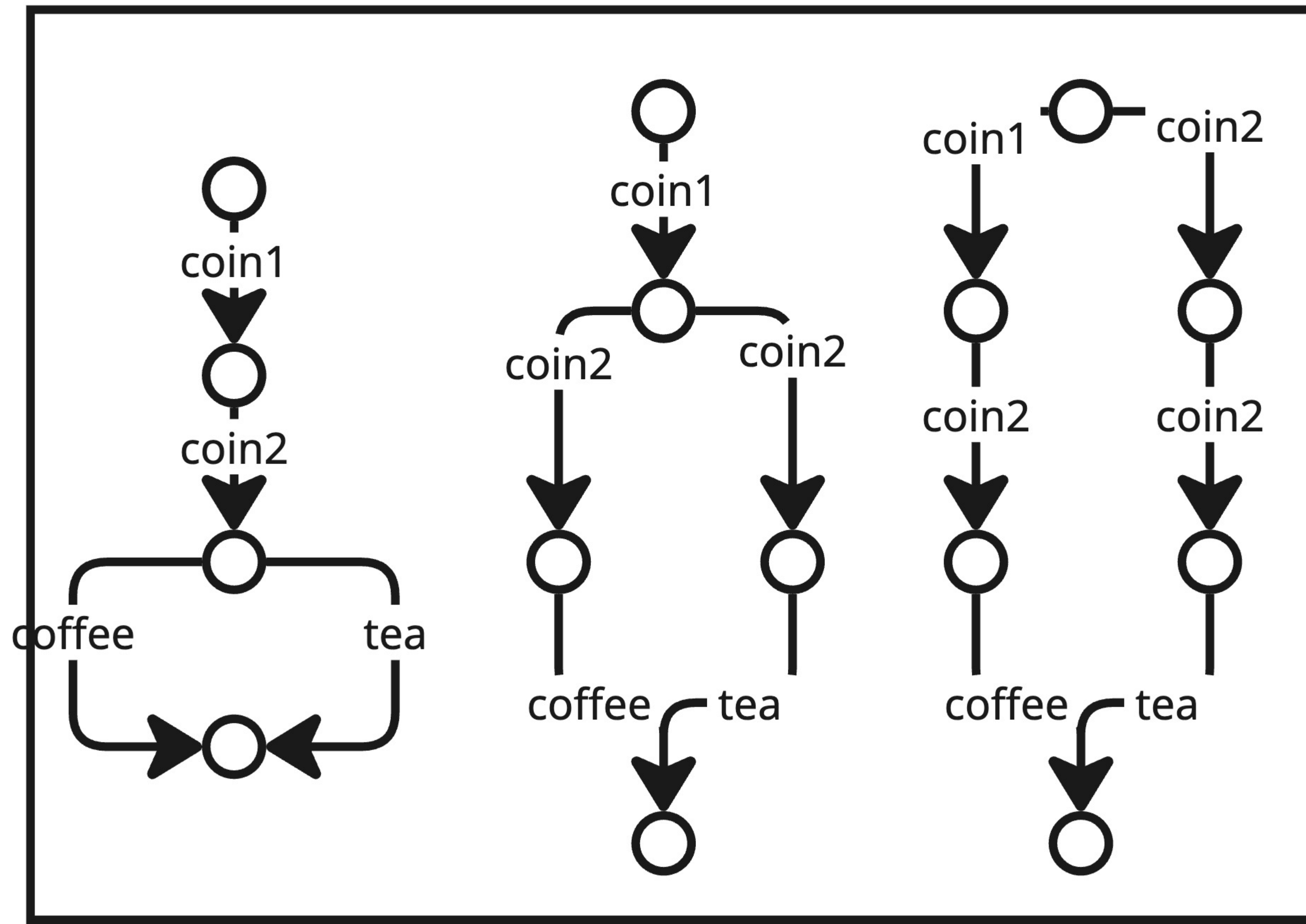
LTS isomorphism may be too restrictive



Trace Equivalences (for now no tau transitions)

Two processes P and Q are trace equivalent:

$$P \sim_{tr} Q \Leftrightarrow \text{traces}(P) = \text{traces}(Q)$$

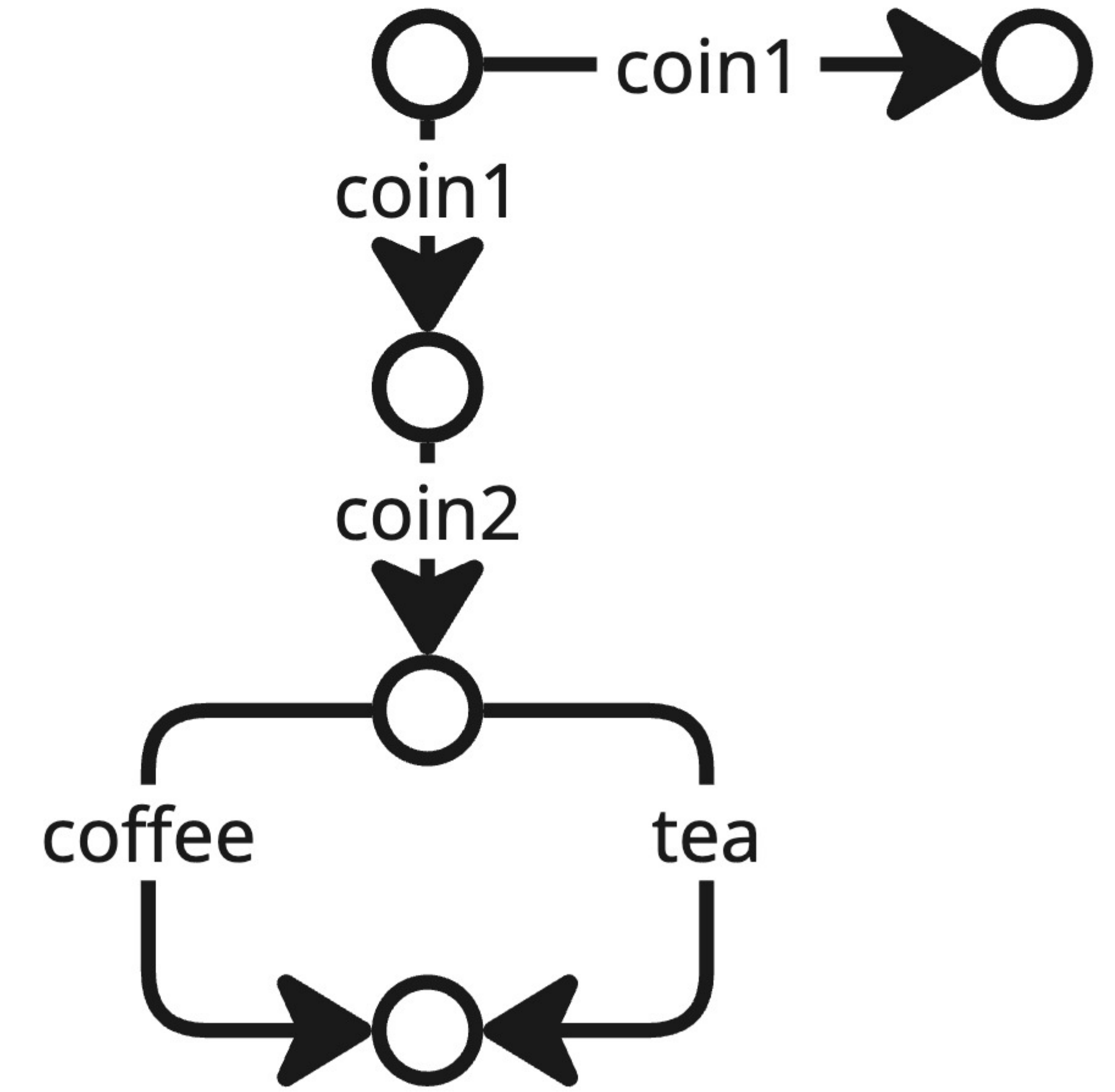
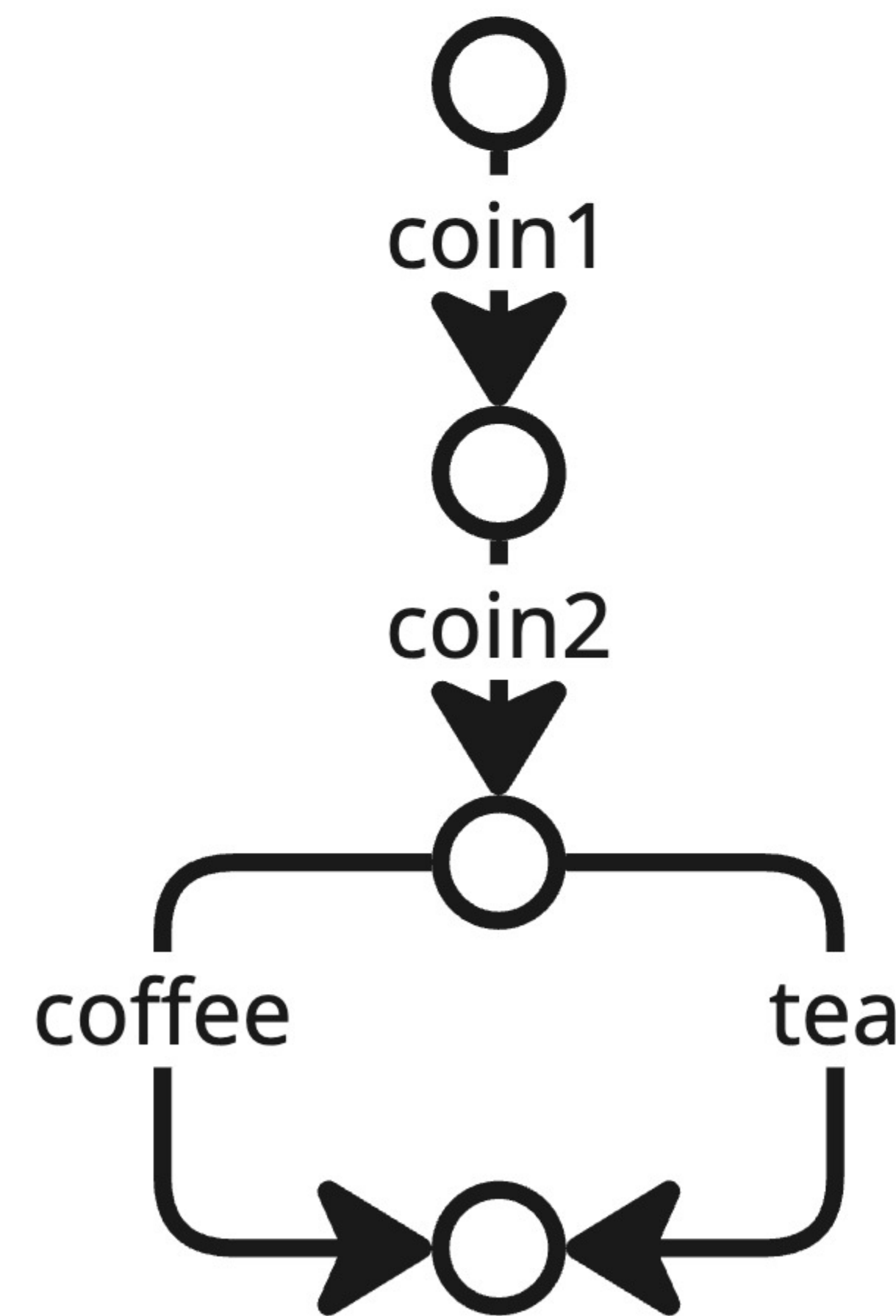


Trace Equivalences (for now no tau transitions)

Two processes P and Q are trace equivalent:

$$P \sim_{tr} Q \Leftrightarrow \text{traces}(P) = \text{traces}(Q)$$

Trace equivalence is not sensitive to deadlocks



Trace Equivalences (for now no tau transitions)

Two processes P and Q are trace equivalent:

$$P \sim_{tr} Q \Leftrightarrow \text{traces}(P) = \text{traces}(Q)$$

$$a.b.0 + a.c.0$$

$$\sim_{tr}$$

$$a.(b.0 + c.0)$$

$$(a.b.0 + a.c.0) \parallel (\bar{a}.\bar{b}.0 + \bar{a}.\bar{c}.0)$$

$$a.(b.0 + c.0) \parallel (\bar{a}.\bar{b}.0 + \bar{a}.\bar{c}.0)$$

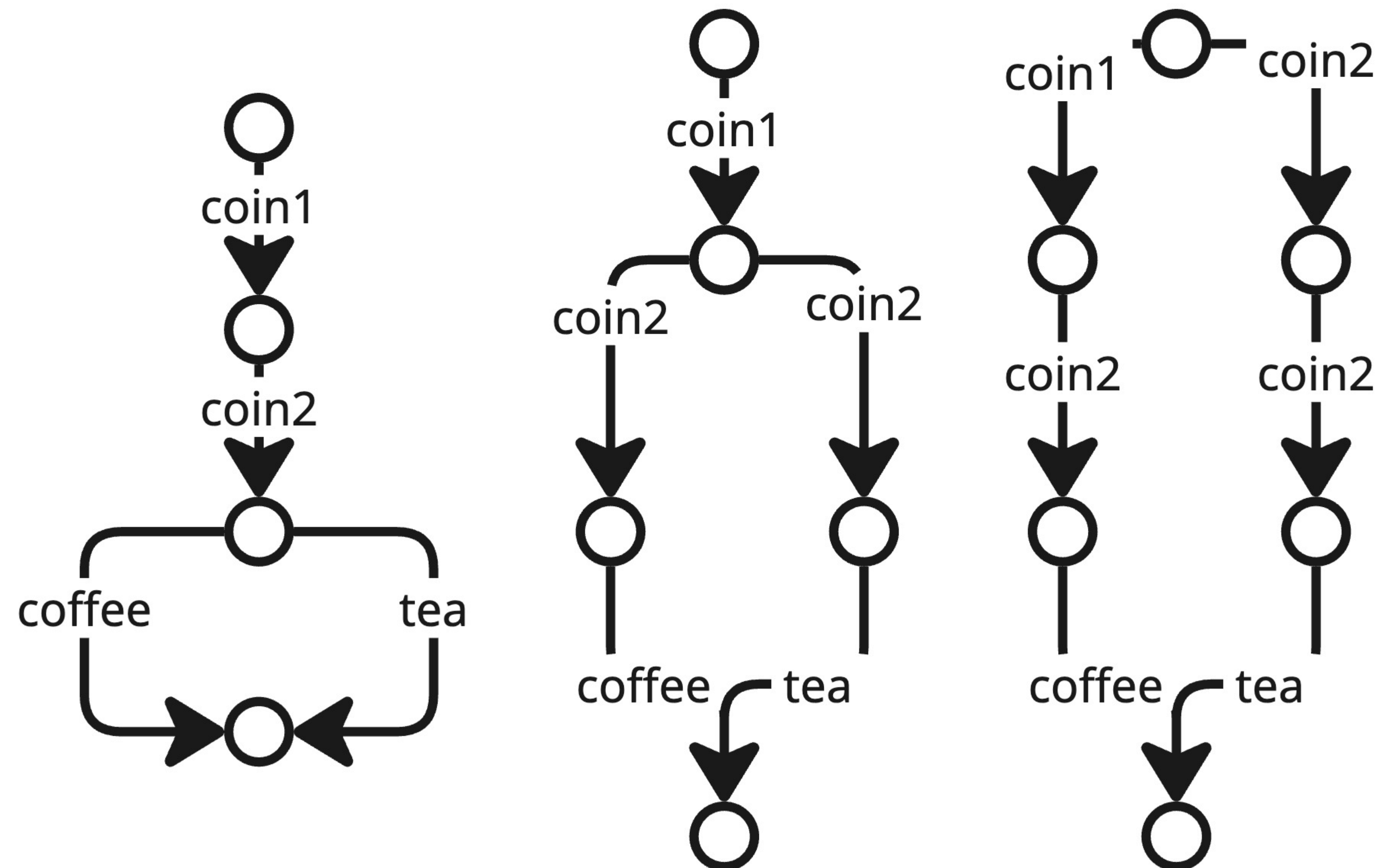
Bisimulation Equivalences

A relation R is a bisimulation if for every $(p, q) \in R$

for every
there is $q \xrightarrow{\alpha} q'$
such that $(p', q') \in R$

for every $q \xrightarrow{\alpha} q'$
there is $p \xrightarrow{\alpha} p'$
such that $(p', q') \in R$

Two states are bisimilar $p \sim q$
if there is a bisimulation R
such that $(p, q) \in R$



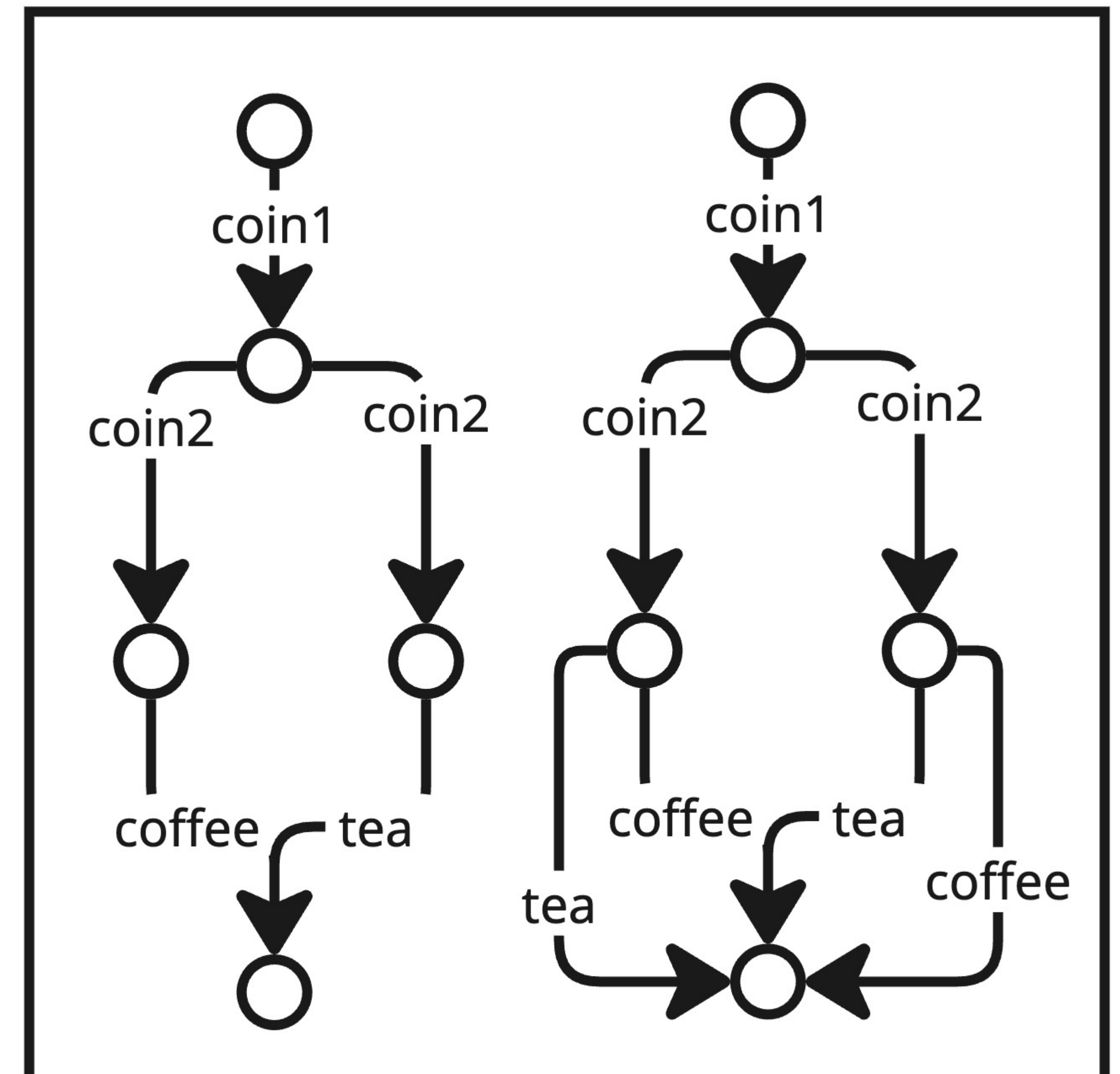
Bisimulation Equivalences

A relation R is a bisimulation if for every $(p, q) \in R$

for every $p \xrightarrow{\alpha} p'$
there is $q \xrightarrow{\alpha} q'$
such that $(p', q') \in R$

for every $q \xrightarrow{\alpha} q'$
there is $p \xrightarrow{\alpha} p'$
such that $(p', q') \in R$

Two states are bisimilar $p \sim q$
if there is a bisimulation R
such that $(p, q) \in R$



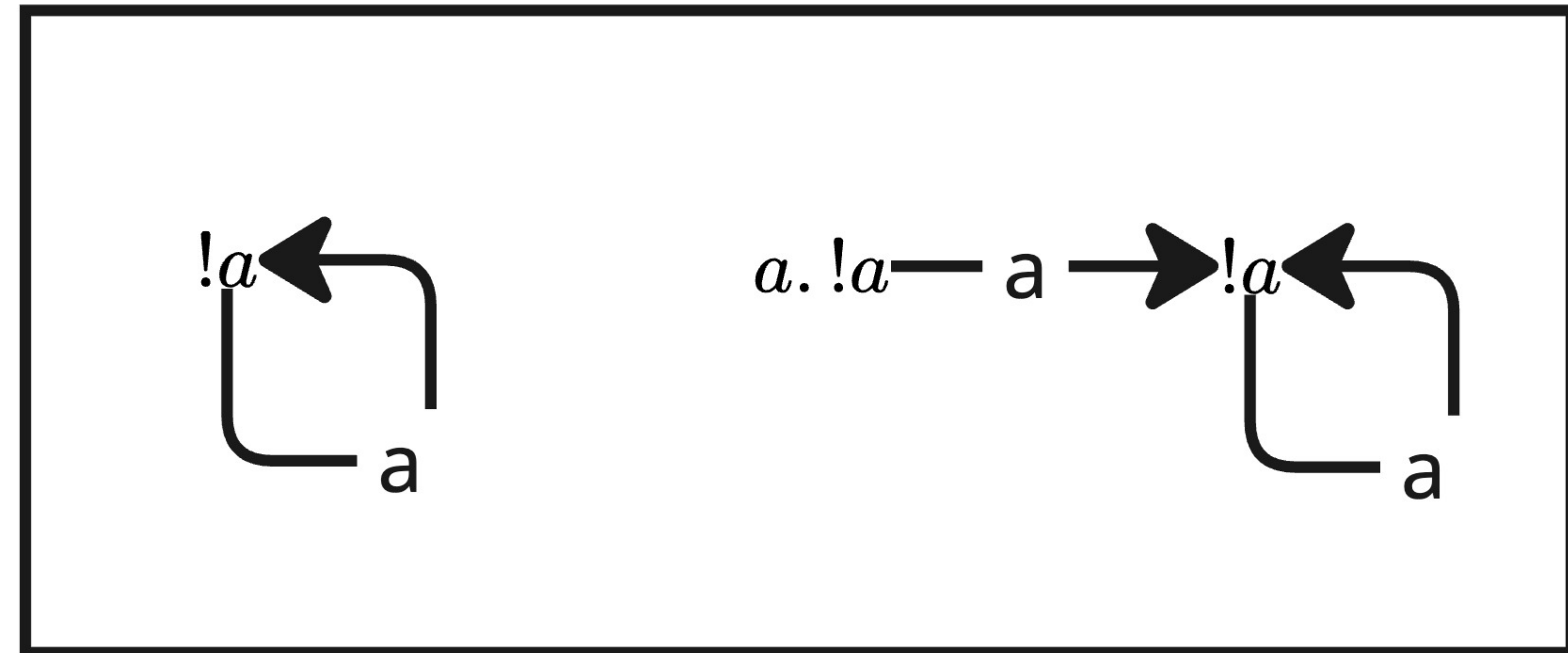
Bisimulation Equivalences

A relation R is a bisimulation if for every $(p, q) \in R$

for every $p \xrightarrow{\alpha} p'$
there is $q \xrightarrow{\alpha} q'$
such that $(p', q') \in R$

for every $q \xrightarrow{\alpha} q'$
there is $p \xrightarrow{\alpha} p'$
such that $(p', q') \in R$

Two states are bisimilar $p \sim q$
if there is a bisimulation R
such that $(p, q) \in R$



Bisimulation Equivalences

A relation R is a bisimulation if for every $(p, q) \in R$

for every

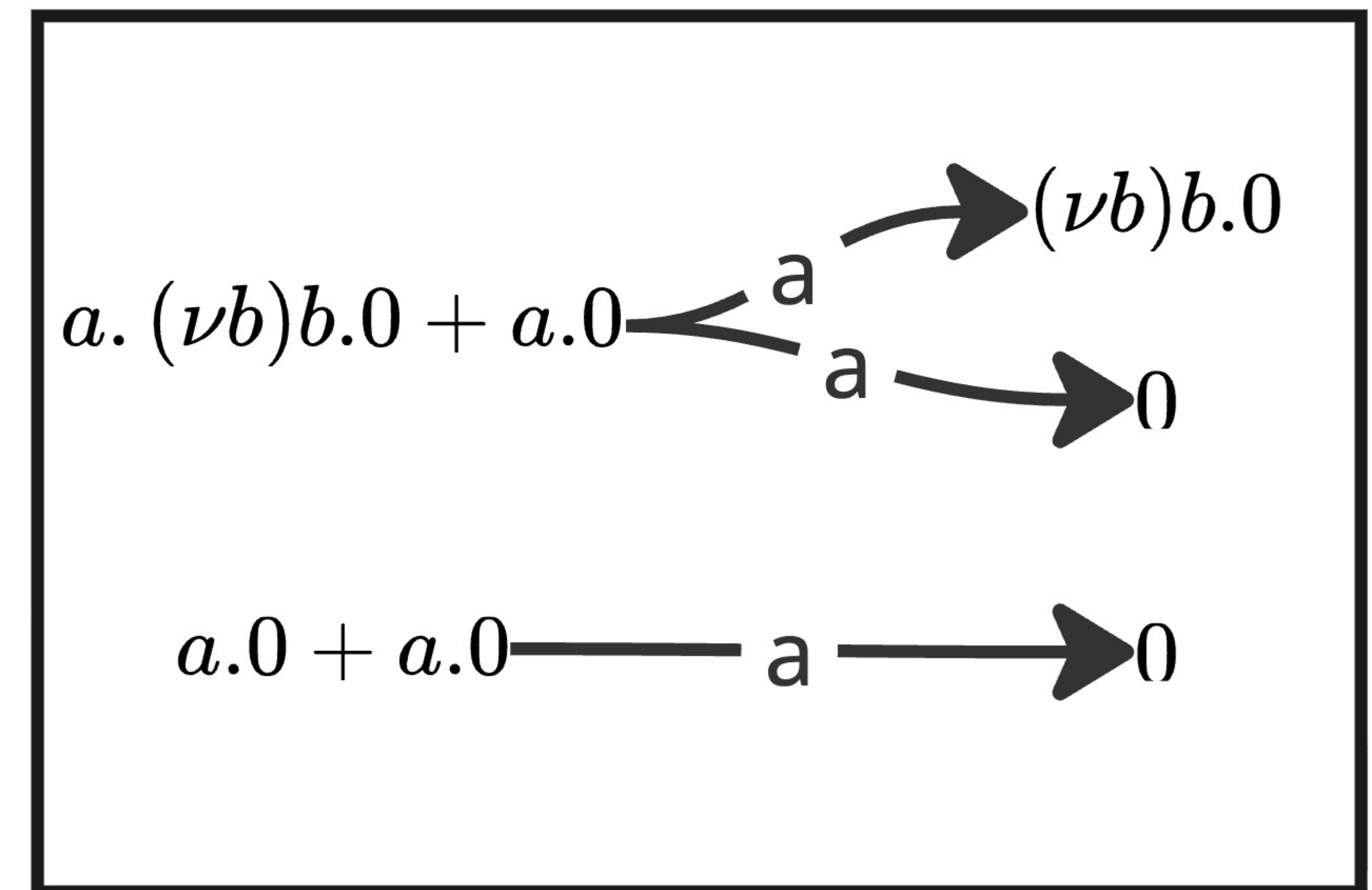
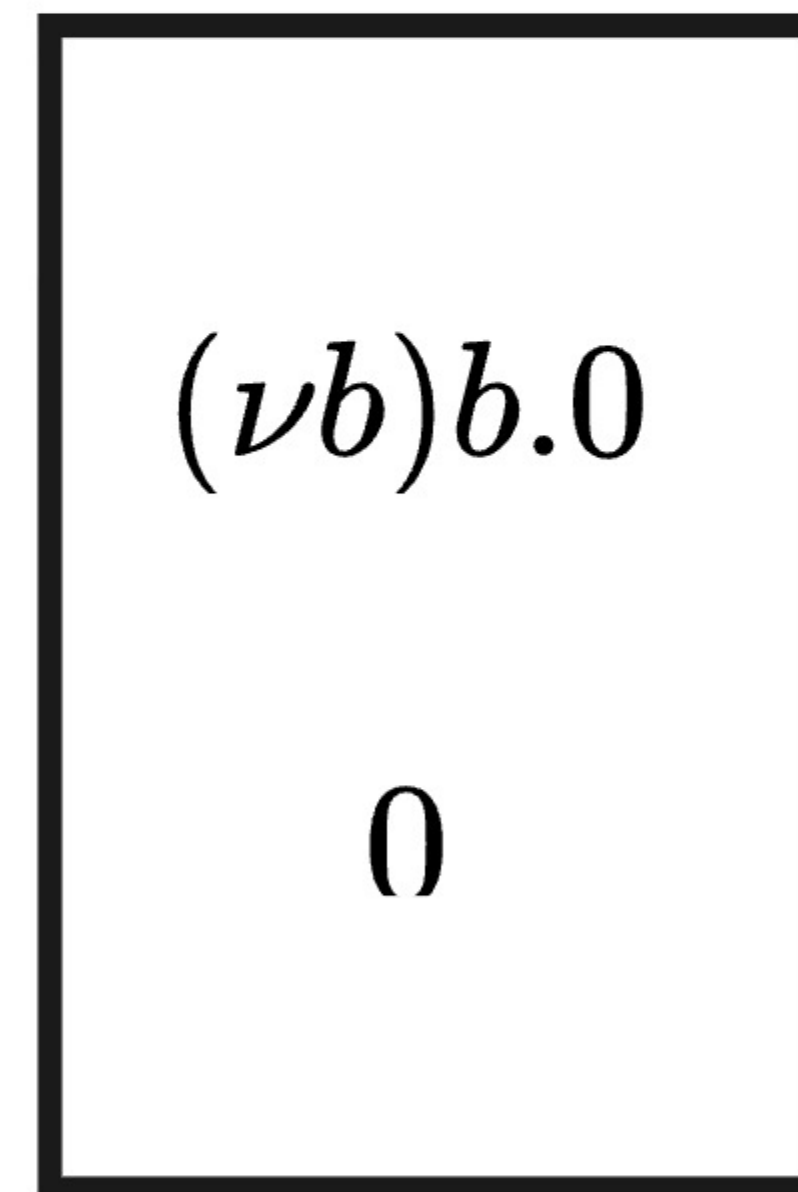
there is $q \xrightarrow{\alpha} q'$
such that $(p', q') \in R$

for every $q \xrightarrow{\alpha} q'$

there is $p \xrightarrow{\alpha} p'$
such that $(p', q') \in R$

Two states are bisimilar $p \sim q$

if there is a bisimulation R
such that $(p, q) \in R$



Bisimulation Equivalence

$$P \parallel Q \sim Q \parallel P$$

$$P \parallel (Q \parallel R) \sim (P \parallel Q) \parallel R$$

$$P \parallel 0 \sim P$$

$$(\nu a)P \sim (\nu a)Q$$

$$a.P \sim a.Q$$

$$P + Q \sim Q + P$$

$$P + (Q + R) \sim (P + Q) + R$$

$$P + 0 \sim P$$

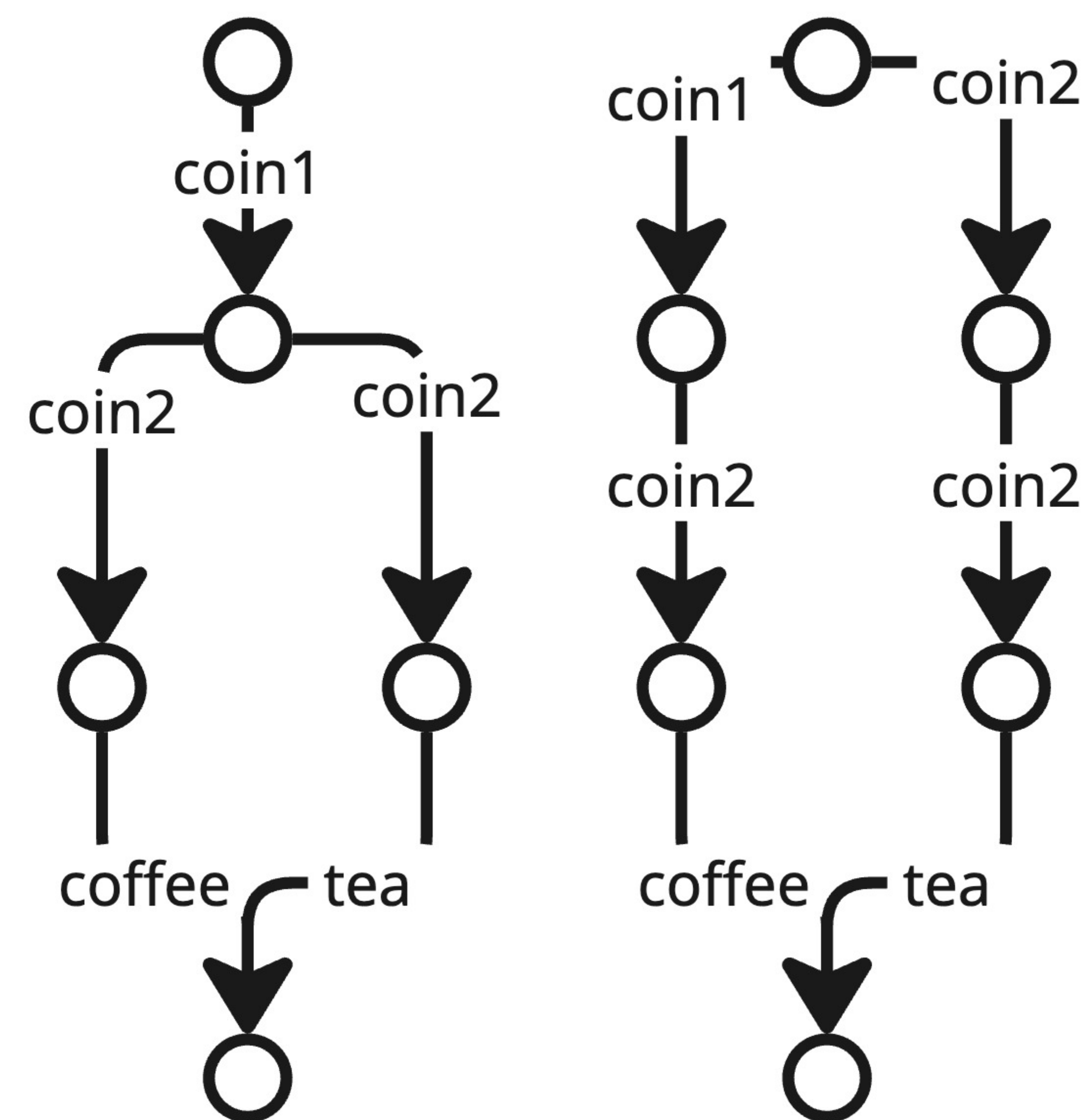
$$P + P \sim P$$

Bisimulation

Bisimulation

- equivalence relation on processes
- is preserved by all syntactic constructors

Bisimulation May still be too restrictive



Testing Equivalences

$$TS_1 = (S_1, Act_1, \rightarrow_1, I_1)$$

$$Test_1 = (S_2, Act_1 \cup w, \rightarrow_2, o), w \notin Act_1$$

$$\frac{p \xrightarrow{\alpha} p' \wedge q \xrightarrow{\alpha} q'}{(p, q) \xrightarrow{\alpha} (p', q')}$$

An observation from (s.o) is a run of the synchronous composition
An observation is successful if contains $(p_n, q_n) \wedge q_n \xrightarrow{w} q'$

Asynchronous π -Calculus

channel names to be communicated along the channels themselves (this is why it is referred to as calculus of mobility)

$$P ::= 0 \mid P \parallel Q \mid P + Q \mid (\nu x)P \mid !P \mid \pi.P$$

$\pi = \tau$ internal actions

$\pi = x(y)$ binds y in the prefixed process. Input a name, refer to it as y , on the channel x

$\pi = \bar{x}y$ does not bind y in the prefixed process. Output a name y on the channel x

Operational semantics π -Calculus

$$fn(0) = \emptyset$$

$$fn(x(y).P) = fn(P) \setminus \{y\} \cup \{x\}$$

$$fn(P \parallel Q) = fn(P) \cup fn(Q)$$

$$fn((\nu x)P) = fn(P) \setminus \{x\}$$

$$fn(P + Q) = fn(P) \cup fn(Q)$$

$$fn(!P) = fn(P)$$

$$fn(\tau.P) = fn(P)$$

$$fn(\bar{x}y.P) = fn(P) \cup \{x, y\}$$

(Early) Operational semantics π -Calculus

labels: $\bar{x}y \mid \bar{x}(y) \mid xy$

$$\begin{array}{c}
 \frac{}{\tau.P \xrightarrow{\tau} P} \qquad \frac{P \xrightarrow{\alpha} P'}{P + Q \xrightarrow{\alpha} P'} \qquad \frac{P \xrightarrow{\alpha} P' \wedge bn(\alpha) \cap fn(Q) = \emptyset}{P \parallel Q \xrightarrow{\alpha} P' \parallel Q} \qquad \frac{P \parallel !P \xrightarrow{\alpha} P'}{!P \xrightarrow{\alpha} P'} \\
 \\
 \frac{}{\bar{x}y.P \xrightarrow{\bar{x}y} P} \qquad \frac{}{x(y).P \xrightarrow{xw} P\{w/y\}} \qquad \frac{P \xrightarrow{\bar{x}y} P' \wedge Q \xrightarrow{xy} Q'}{P \parallel Q \xrightarrow{\tau} P' \parallel Q'} \\
 \\
 \frac{P \xrightarrow{\bar{x}y} P' \wedge y \neq x \wedge w \notin fn((\nu y)P')}{(\nu y)P \xrightarrow{\bar{x}(w)} P'} \qquad \frac{P \xrightarrow{\bar{x}(y)} P' \wedge Q \xrightarrow{xy} Q'}{P \parallel Q \xrightarrow{\tau} (\nu y)(P' \parallel Q')} \qquad \frac{P \xrightarrow{\alpha} P' \wedge x \notin n(\alpha)}{(\nu x)P \xrightarrow{\alpha} (\nu x)P'}
 \end{array}$$


(Late) Operational semantics π -Calculus

labels: $\bar{x}y \mid \bar{x}(y) \mid xy$

$$\begin{array}{c} \frac{}{\tau.P \xrightarrow{\tau} P} \quad \frac{P \xrightarrow{\alpha} P'}{P + Q \xrightarrow{\alpha} P'} \quad \frac{P \xrightarrow{\alpha} P' \wedge bn(\alpha) \cap fn(Q) = \emptyset}{P \parallel Q \xrightarrow{\alpha} P' \parallel Q} \quad \frac{P \parallel !P \xrightarrow{\alpha} P'}{!P \xrightarrow{\alpha} P'} \\[10pt] \frac{}{\bar{x}y.P \xrightarrow{\bar{x}y} P} \quad \frac{}{x(y).P \xrightarrow{xy} P\{w/y\}} \quad \frac{P \xrightarrow{\bar{x}w} P' \wedge Q \xrightarrow{xy} Q'}{P \parallel Q \xrightarrow{\tau} P' \parallel Q'\{w/y\}} \\[10pt] \frac{P \xrightarrow{\bar{x}y} P' \wedge y \neq x \wedge w \notin fn((\nu y)P')}{(\nu y)P \xrightarrow{\bar{x}(w)} P'} \quad \frac{P \xrightarrow{\bar{x}(y)} P' \wedge Q \xrightarrow{xy} Q'}{P \parallel Q \xrightarrow{\tau} (\nu y)(P' \parallel Q')} \quad \frac{P \xrightarrow{\alpha} P' \wedge x \notin n(\alpha)}{(\nu x)P \xrightarrow{\alpha} (\nu x)P'} \end{array}$$