http://didawiki.di.unipi.it/doku.php/magistraleinformatica/psc/start

PSC 2022/23 (375AA, 9CFU)

Principles for Software Composition

Roberto Bruni
http://www.di.unipi.it/~bruni/

27 - PEPA

PEPA
Performance Evaluation Process Algebra

Building models

Conceptualise your system as a Markov chain

Construct your Markov chain (infinitesimal generator matrix)

Solve your equations to derive quantitative information

Building models

Conceptualise your system as a Markov chain

Construct your Markov chain (infinitesimal generator matrix)

Solve your equations to derive quantitative information

Monolithic approach: not suitable for complex systems

PEPA project

the PEPA project started in Edinburgh in 1991

motivated by the performance analysis
of large computer and communication systems

exploit interplay between Process Algebras and CTMC

Process Algebras (PA):
compositional description of complex systems,
formal reasoning (for correctness)

CTMC:
numerical analysis

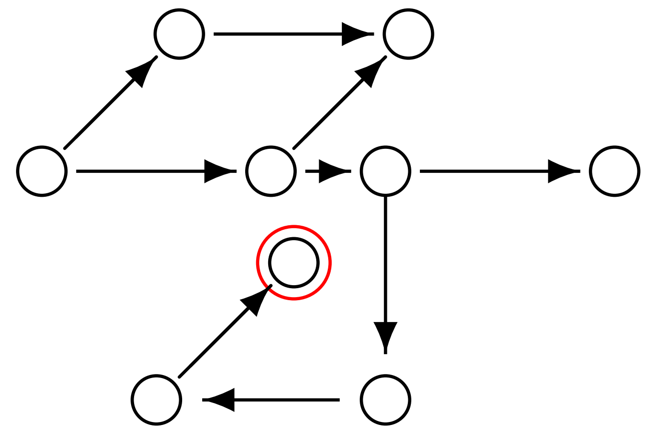
compositional construction of CTMC

PEPA meets CTMC

PA	mutual influence	CTMC
interaction designed around CTMC		ease of construction
actions have durations		design of independent components
add rates to labels		cooperation between components
probabilistic branching		explicit interaction
quantitative measures		reusable sub-models
probabilistic model checking		easy to understand models
quantitative logics		space reduction techniques
		functional verification

Formal models

qualitative vs quantitative



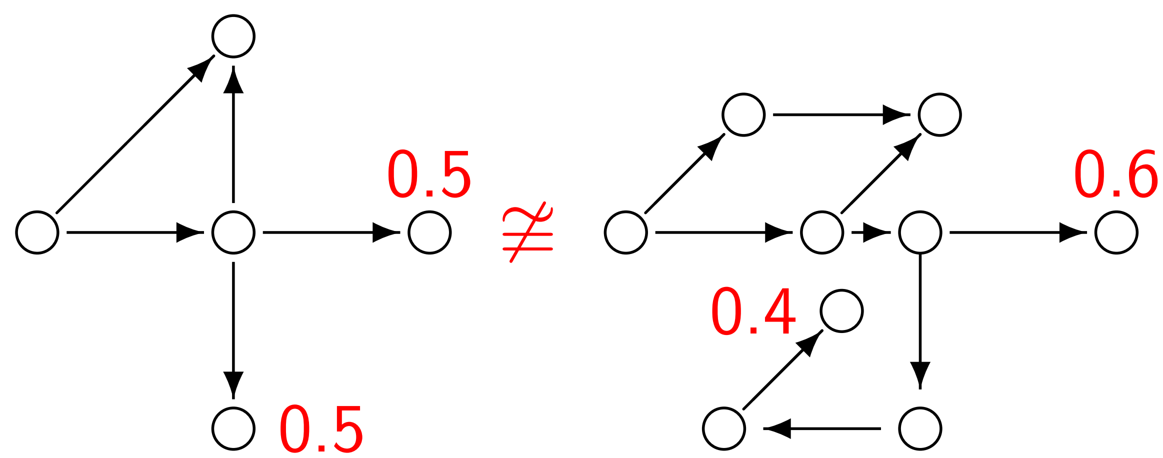
reachability:
will the system arrive to a particular state?

how long will it take the system to arrive to a particular state?

(taken from Jane Hillston's slides)

Formal models

qualitative vs quantitative



conformance:
does system behaviour match its specification?

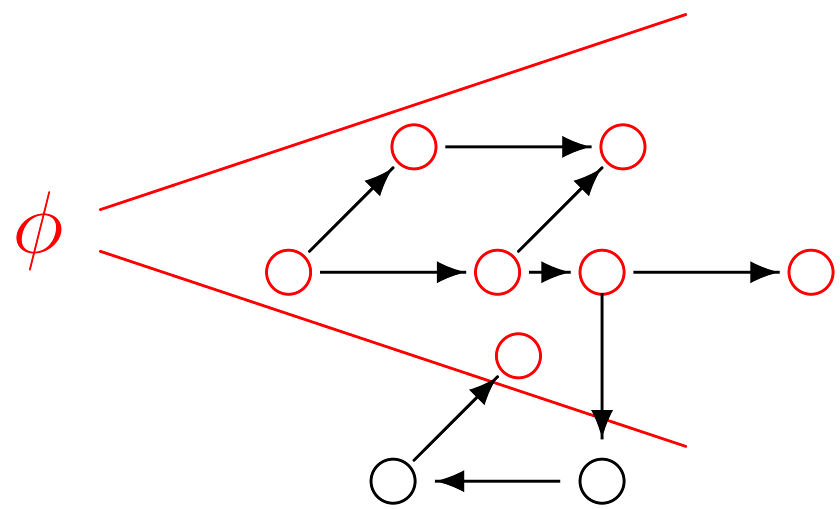
how likely is that system behaviour will match its specification?

does the frequency profile of the system match that of its specification?

(taken from Jane Hillston's slides)

Formal models

qualitative vs quantitative



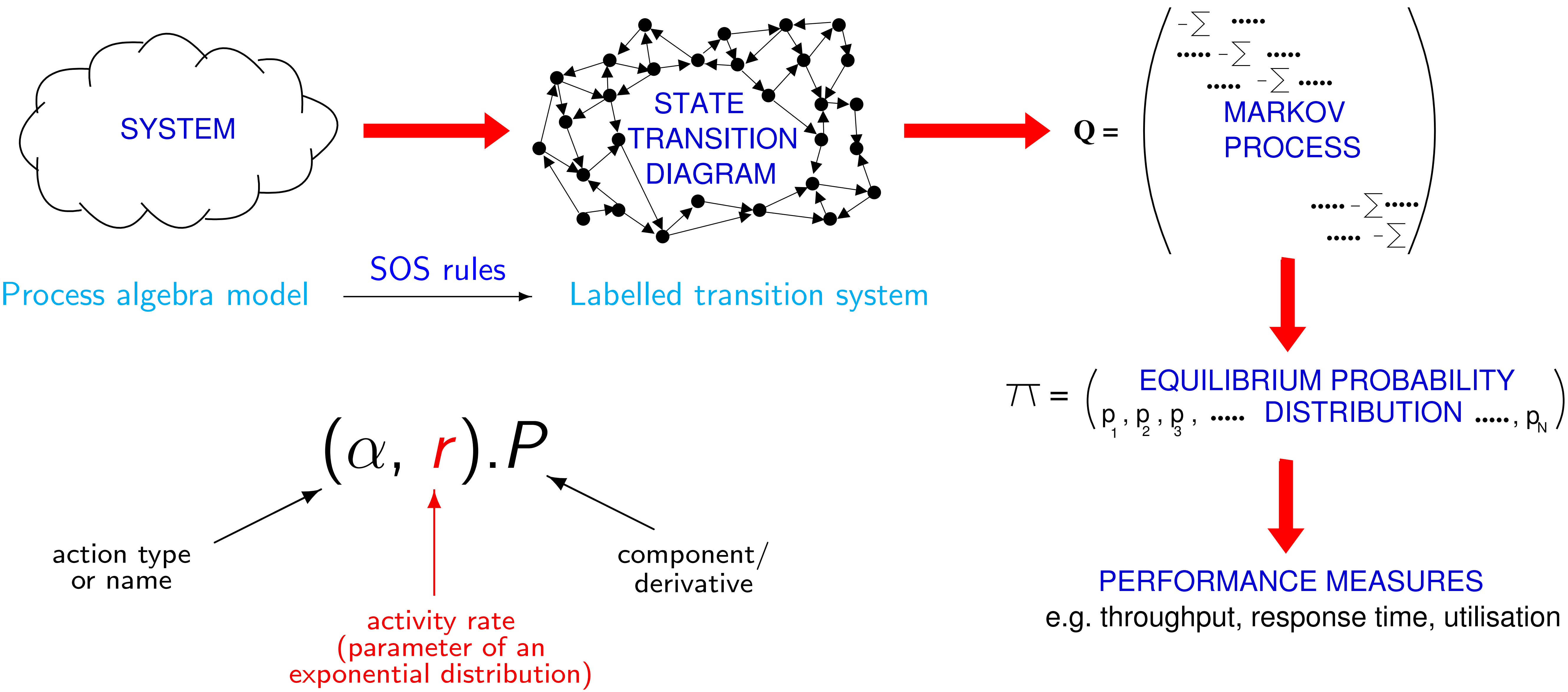
verification:
does a given property
hold within the system?

Does a given property
hold within the system
with a given probability?

How long is it until
a given probability hold?

(taken from Jane Hillston's slides)

PEPA workflow



(taken from Jane Hillston's slides)

Communication style

PEPA parallel composition is based on Hoare's CSP

CCS-style	CSP-style
actions and co-actions	no i/o distinction
binary synchronisation	multiple cooperation
conjugate sync	shared name sync
result in a silent action	result in the same name
restriction	hiding
parallel composition	cooperation combinator
one operator	parametric operator

CSP cooperation combinator

$$P \underset{L}{\bowtie} Q$$

cooperation set

interleaving

$$\frac{P_1 \xrightarrow{\alpha} Q_1 \quad \boxed{\alpha \notin L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{\alpha} Q_1 \underset{L}{\bowtie} P_2} \qquad \frac{P_2 \xrightarrow{\alpha} Q_2 \quad \boxed{\alpha \notin L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{\alpha} P_1 \underset{L}{\bowtie} Q_2}$$

cooperation

$$\frac{P_1 \xrightarrow{\alpha} Q_1 \quad P_2 \xrightarrow{\alpha} Q_2 \quad \boxed{\alpha \in L}}{P_1 \underset{L}{\bowtie} P_2 \boxed{\xrightarrow{\alpha}} Q_1 \underset{L}{\bowtie} Q_2}$$

pure interleaving

$$P \parallel Q \triangleq P \underset{\emptyset}{\bowtie} Q$$

PEPA
syntax and semantics

PEPA syntax

P, Q	$::=$	$\mathbf{nil}$	inactive process
	$\mid$	$(\alpha, r).P$	action prefix
	$\mid$	$P + Q$	choice
	$\mid$	$P \underset{L}{\bowtie} Q$	cooperation combinator
	$\mid$	P/L	hiding
	$\mid$	C	process constant

$\alpha \in \Lambda$ action

$L \subseteq \Lambda$ set of actions

$\Delta = \{C_i \triangleq P_i\}_{i \in I}$ set of process declarations

PEPA LTS

ongoing interaction
with the environment
(with other processes)
and its rate

$$P \xrightarrow{(\alpha, r)} Q$$

a process
in its current
state

the process
state after the
interaction

small-step semantics

PEPA semantics (basics)

$$\frac{}{(\alpha, r).P \xrightarrow{(\alpha,r)} P}$$

$$\frac{P_1 \xrightarrow{(\alpha,r)} Q}{P_1 + P_2 \xrightarrow{(\alpha,r)} Q} \qquad \frac{P_2 \xrightarrow{(\alpha,r)} Q}{P_1 + P_2 \xrightarrow{(\alpha,r)} Q}$$

$$\frac{C \triangleq P \in \Delta \quad P \xrightarrow{(\alpha,r)} Q}{C \xrightarrow{(\alpha,r)} Q}$$

Example

$$\text{Server} \quad \stackrel{\Delta}{=} \quad (get, \top).(download, \mu).(rel, \top).\text{Server}$$

extremely high rate
cannot influence the overall rate
of interacting components

$$\begin{aligned} \text{Browser} \quad & \stackrel{\Delta}{=} \quad (display, \lambda_1).(cache, m).\text{Browser} \\ & + \quad (display, \lambda_2).(get, g).(download, \top).(rel, r).\text{Browser} \end{aligned}$$

a local choice
taken with probability $\frac{\lambda_i}{\lambda_1+\lambda_2}$

Hiding and interleaving

$$\frac{P \xrightarrow{(\alpha,r)} Q \quad \boxed{\alpha \notin L}}{P/L \xrightarrow{(\alpha,r)} Q/L}$$

$$\frac{P \xrightarrow{(\alpha,r)} Q \quad \boxed{\alpha \in L}}{P/L \boxed{\xrightarrow{(\tau,r)}} Q/L}$$

$$\frac{P_1 \xrightarrow{(\alpha,r)} Q_1 \quad \boxed{\alpha \notin L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r)} Q_1 \underset{L}{\bowtie} P_2}$$

$$\frac{P_2 \xrightarrow{(\alpha,r)} Q_2 \quad \boxed{\alpha \notin L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r)} P_1 \underset{L}{\bowtie} Q_2}$$

Cooperation

$$\frac{P_1 \xrightarrow{(\alpha,r_1)} Q_1 \quad P_2 \xrightarrow{(\alpha,r_2)} Q_2 \quad \boxed{\alpha \in L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r)} Q_1 \underset{L}{\bowtie} Q_2}$$

which rate should we put here?

Multiway synchronization

$$F \stackrel{def}{=} (fork, r_f).(join, r_j).F'$$

$$W_1 \stackrel{def}{=} (fork, r_{f_1}).(doWork_1, r_1).W_1'$$

$$W_2 \stackrel{def}{=} (fork, r_{f_2}).(doWork_2, r_2).W_2'$$

$$F' \stackrel{def}{=} \ldots, \quad W_1' \stackrel{def}{=} \ldots, \quad W_2' \stackrel{def}{=} \ldots$$

$$System \stackrel{def}{=} (F \underset{\{fork\}}{\bowtie} W_1) \underset{\{fork\}}{\bowtie} W_2$$

$$\cfrac{\cfrac{F \xrightarrow{(fork, r_f)} (join, r_j)F' \qquad W_1 \xrightarrow{(fork, r_{f_1})} (doWork_1, r_1).W_1'}{F \underset{\{fork\}}{\bowtie} W_1 \xrightarrow{(fork, r')} (join, r_j).F' \underset{\{fork\}}{\bowtie} (doWork_1, r_1).W_1'} \qquad W_2 \xrightarrow{(fork, r_{f_2})} (doWork_2, r_2).W_2'}{F \underset{\{fork\}}{\bowtie} W_1 \underset{\{fork\}}{\bowtie} W_2 \xrightarrow{(fork, r'')} (join, r_j).F' \underset{\{fork\}}{\bowtie} (doWork_1, r_1)W_1' \underset{\{fork\}}{\bowtie} (doWork_2, r_2).W_2'}$$

(taken from Mirco Tribastone's slides)

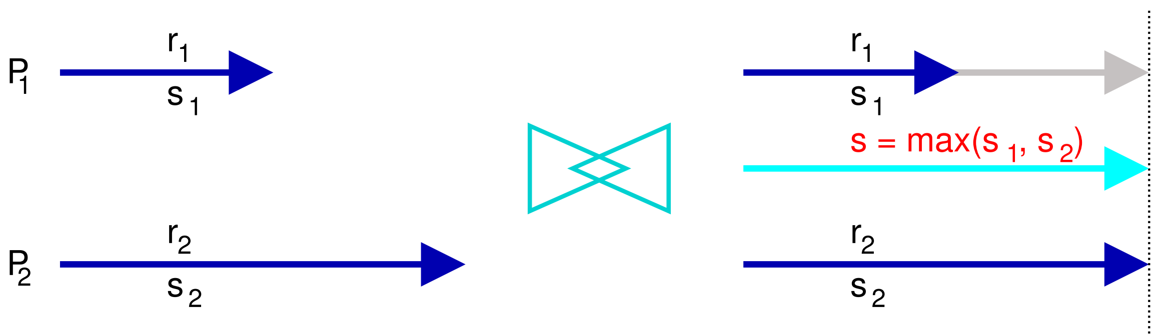
Exclusive cooperation

$$\begin{aligned} Premium &\stackrel{\mathit{def}}{=} (dwn, r_p).Premium' \\ Basic &\stackrel{\mathit{def}}{=} (dwn, r_b).Basic' \\ S &\stackrel{\mathit{def}}{=} (dwn, r_s).S' \\ &\cdots \\ System &\stackrel{\mathit{def}}{=} (Premium \parallel Basic) \underset{L}{\bowtie} S, \\ L &= \{dwn\} \end{aligned}$$

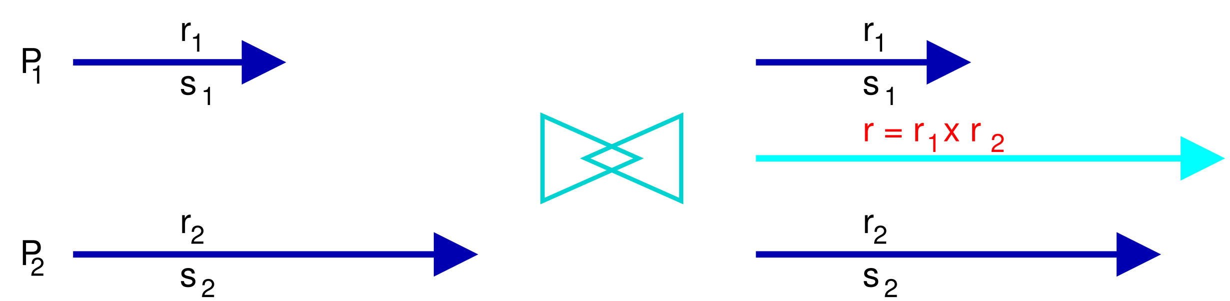
$$\dfrac{\dfrac{Premium \xrightarrow{(dwn,r_p)} Premium'}{Premium \parallel Basic \xrightarrow{(dwn,r_p)} Premium' \parallel Basic} \qquad \dfrac{}{S \xrightarrow{(dwn,r_s)} S'}}{\dfrac{Premium \parallel Basic \underset{L}{\bowtie} S \xrightarrow{(dwn,{\color{red}r_{ps}})} Premium' \parallel Basic \underset{L}{\bowtie} S'}{{\color{red} System \xrightarrow{(dwn,r_{ps})} Premium' \parallel Basic \underset{L}{\bowtie} S'}}}$$

Which rate for sync?

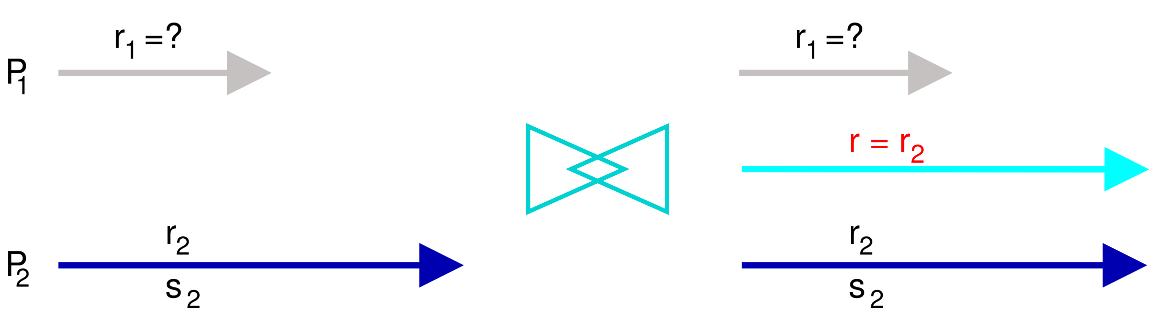
stochastic PA differ for the treatment of rates of synchronised actions



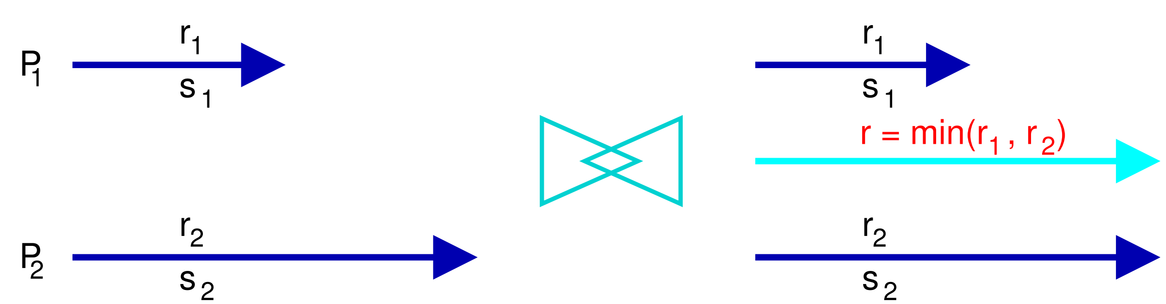
s is no longer exponentially distributed



TIPP: new rate is product of individual rates



EMPA: one participant is passive



bounded capacity: new rate is the minimum of the rates

PEPA's approach

(taken from Jane Hillston's slides)

PEPA: bounded capacity

Each component has a bounded capacity to carry out activities of some type, determined by the apparent rate for that type

cooperation cannot make a component exceed its bounded capacity

thus the apparent rate of a cooperation is the minimum of the apparent rates of the co-operands

PEPA: apparent rates

No component can be made to carry out an action in cooperation faster than its own defined rate for the actions

thus shared actions proceed at the minimum of the rates in the participating components

the apparent rates of independent actions is instead the sum of their rates within independent concurrent components

PEPA: apparent rate

$r_\alpha(P)$ is the observed rate of action α in P

$$
\begin{tikzcd}
& P \arrow[dl, "(\alpha{,}r_1)"'] \arrow[dr, "(\alpha{,}r_n)"] & \\
Q_1 & \cdots & Q_n
\end{tikzcd}
$$

$$r_\alpha(P) = r_1 + \cdots + r_n$$

PEPA: apparent rate

$r_\alpha(P)$ is the observed rate of action α in P

$$r_\alpha(\mathbf{nil}) \triangleq 0$$

$$r_\alpha((\beta, r).P) \triangleq \begin{cases} r & \text{if } \alpha = \beta \\ 0 & \text{if } \alpha \neq \beta \end{cases}$$

$$r_\alpha(P + Q) \triangleq r_\alpha(P) + r_\alpha(Q)$$

(+ is not idempotent!)

$$r_\alpha(P/L) \triangleq \begin{cases} r_\alpha(P) & \text{if } \alpha \notin L \\ 0 & \text{if } \alpha \in L \end{cases}$$

actions are interleaved

$$r_\alpha(P \underset{L}{\bowtie} Q) \triangleq \begin{cases} r_\alpha(P) + r_\alpha(Q) & \text{if } \alpha \notin L \\ \min\{r_\alpha(P), r_\alpha(Q)\} & \text{if } \alpha \in L \end{cases}$$

the slowest must be waited for

$$r_\alpha(C) \triangleq r_\alpha(P) \quad \text{if } C \triangleq P \in \Delta$$

Cooperation

$$\frac{P_1 \xrightarrow{(\alpha,r_1)} Q_1 \quad P_2 \xrightarrow{(\alpha,r_2)} Q_2 \quad \boxed{\alpha \in L}}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r)} Q_1 \underset{L}{\bowtie} Q_2}$$

$$r = r_\alpha(P_1 \underset{L}{\bowtie} P_2) \cdot \frac{r_1}{r_\alpha(P_1)} \cdot \frac{r_2}{r_\alpha(P_2)}$$

apparent rate

the sum of the rates of all the α-transitions that $P_1 \underset{L}{\bowtie} P_2$ can do

probability of specific action (α, r_i) among the α-transitions of P_i

Cooperation: example

For r_1, r_2 positive reals,

$$\frac{(\alpha, r_1).P_1 \xrightarrow{(\alpha,r_1)} P_1 \qquad (\alpha, r_2).P_2 \xrightarrow{(\alpha,r_2)} P_2}{(\alpha, r_1).P_1 \underset{\{\alpha\}}{\bowtie} (\alpha, r_2).P_2 \xrightarrow{(\alpha,R)} P_1 \underset{\{\alpha\}}{\bowtie} P_2},$$

where

$$\begin{aligned} R &= \frac{r_1}{r_\alpha\big((\alpha, r_1).P_1\big)} \frac{r_2}{r_\alpha\big((\alpha, r_2).P_2\big)} \min\Big(r_\alpha\big((\alpha, r_1).P_1\big), r_\alpha\big((\alpha, r_2).P_2\big)\Big) \\ &= \frac{r_1}{r_1}\frac{r_2}{r_2}\min(r_1, r_2) = \min(r_1, r_2). \end{aligned}$$

We recover the intuitive definition of the minimum between the two rates.

(taken from Mirco Tribastone's slides)

Cooperation: example

For r a positive real,

$$\frac{(\alpha, r).P_1 \xrightarrow{(\alpha,r)} P_1 \qquad (\alpha, \top).P_2 \xrightarrow{(\alpha,\top)} P_2}{(\alpha, r).P_1 \underset{\{\alpha\}}{\bowtie} (\alpha, \top).P_2 \xrightarrow{(\alpha,R)} P_1 \underset{\{\alpha\}}{\bowtie} P_2},$$

where

$$\begin{aligned} R &= \frac{r}{r_\alpha\big((\alpha, r).P_1\big)} \frac{\top}{r_\alpha\big((\alpha, \top).P_2\big)} \min\Big(r_\alpha\big((\alpha, r).P_1\big), r_\alpha\big((\alpha, \top).P_2\big)\Big) \\ &= \frac{r}{r}\frac{\top}{\top}\min(r, \top) = r. \end{aligned}$$

We recover the intuitive definition of infinite capacity — the rate of synchronisation is determined by the active component.

(taken from Mirco Tribastone's slides)

Apparent rates in active cooperation

$$\boxed{\begin{aligned} Cli &\stackrel{def}{=} (\alpha, r_d).Cli' \\ Ser &\stackrel{def}{=} (\alpha, r_u).Ser' \\ Sys &\stackrel{def}{=} (Cli \parallel Cli) \underset{\{\alpha\}}{\bowtie} Ser \end{aligned}}$$

$$\cfrac{\cfrac{\cfrac{}{(\alpha, r_d).Cli' \xrightarrow{(\alpha, r_d)} Cli'}}{Cli \xrightarrow{(\alpha, r_d)} Cli'}}{Cli \parallel Cli \xrightarrow{(\alpha, r_d)} Cli' \parallel Cli} \qquad \cfrac{(\alpha, r_u).Ser' \xrightarrow{(\alpha, r_u)} Ser'}{Ser \xrightarrow{(\alpha, r_u)} Ser'}$$

$$\frac{Cli \parallel Cli \xrightarrow{(\alpha, r_d)} Cli' \parallel Cli \qquad Ser \xrightarrow{(\alpha, r_u)} Ser'}{Cli \parallel Cli \underset{\{\alpha\}}{\bowtie} Ser \xrightarrow{(\alpha, R')} Cli' \parallel Cli \underset{\{\alpha\}}{\bowtie} Ser'},$$

$$R' = \frac{r_d}{r_d + r_d} \frac{r_u}{r_u} \min(r_d + r_d, r_u) = \frac{1}{2} \min(r_d + r_d, r_u)$$

(taken from Mirco Tribastone's slides)

Apparent rates in active cooperation

$$\begin{aligned} Cli &\overset{def}{=} (\alpha, r_d).Cli' \\ Ser &\overset{def}{=} (\alpha, r_u).Ser' \\ Sys &\overset{def}{=} (Cli \parallel Cli) \underset{\{\alpha\}}{\bowtie} Ser \end{aligned}$$

$$\frac{\dfrac{\dfrac{}{(\alpha, r_d).Cli' \xrightarrow{(\alpha, r_d)} Cli'}}{Cli \xrightarrow{(\alpha, r_d)} Cli'}}{\color{red}{Cli \parallel Cli \xrightarrow{(\alpha, r_d)} Cli \parallel Cli'}} \quad \frac{(\alpha, r_u).Ser' \xrightarrow{(\alpha, r_u)} Ser'}{Ser \xrightarrow{(\alpha, r_u)} Ser'}$$

$$\frac{\color{red}{Cli \parallel Cli \xrightarrow{(\alpha, r_d)} Cli \parallel Cli'} \qquad Ser \xrightarrow{(\alpha, r_u)} Ser'}{Cli \parallel Cli \underset{\{\alpha\}}{\bowtie} Ser \xrightarrow{(\alpha, R'')} Cli \parallel Cli' \underset{\{\alpha\}}{\bowtie} Ser'},$$

$$R'' = \frac{r_d}{r_d + r_d} \frac{r_u}{r_u} \min(r_d + r_d, r_u) = \frac{1}{2} \min(r_d + r_d, r_u) = R'$$

(taken from Mirco Tribastone's slides)

Apparent rates in active cooperation

$$\boxed{\begin{aligned} Cli &\stackrel{def}{=} (\alpha, r_d).Cli' \\ Ser &\stackrel{def}{=} (\alpha, r_u).Ser' \\ Sys &\stackrel{def}{=} (Cli \parallel Cli) \underset{\{\alpha\}}{\bowtie} Ser \end{aligned}}$$

$$\begin{tikzcd} & Cli \parallel Cli \underset{\{\alpha\}}{\bowtie} Ser \arrow[dl, "{\big(\alpha, 1/2\min(2\,r_d, r_u)\big)}"'] \arrow[dr, "{\big(\alpha, 1/2\min(2\,r_d, r_u)\big)}"] & \\ Cli' \parallel Cli \underset{\{\alpha\}}{\bowtie} Ser' & & Cli \parallel Cli' \underset{\{\alpha\}}{\bowtie} Ser' \end{tikzcd}$$

(taken from Mirco Tribastone’s slides)

Careful with that cooperation set

- $\left((\alpha, r).P \underset{\{\alpha\}}{\bowtie} (\alpha, s).Q\right) \underset{\{\alpha\}}{\bowtie} (\alpha, t).R$
- $((\alpha, r).P \parallel (\alpha, s).Q) \underset{\{\alpha\}}{\bowtie} (\alpha, t).R$
- $\left((\alpha, r).P \underset{\{\alpha\}}{\bowtie} (\alpha, s).Q\right) \parallel (\alpha, t).R$

(taken from Jane Hillston's slides)

Example

$$\begin{aligned}
\text{Server} &\triangleq (get, \top).(download, \mu).(rel, \top).\text{Server} \\
\text{S} &\triangleq (get, \top).\text{S1} \\
\text{S1} &\triangleq (dnd, \mu).\text{S2} \\
\text{S2} &\triangleq (rel, \top).\text{S}
\end{aligned}$$

$$\begin{aligned}
\text{Browser} &\triangleq (display, \lambda_1).(cache, m).\text{Browser} \\
&+ (display, \lambda_2).(get, g).(download, \top).(rel, r).\text{Browser} \\
\text{B} &\triangleq (dis, \lambda_1).\text{B1} + (dis, \lambda_2).\text{B2} \\
\text{B1} &\triangleq (cac, m).\text{B} \\
\text{B2} &\triangleq (get, g).\text{B3} \\
\text{B3} &\triangleq (dnd, \top).\text{B4} \\
\text{B4} &\triangleq (rel, r).\text{B}
\end{aligned}$$

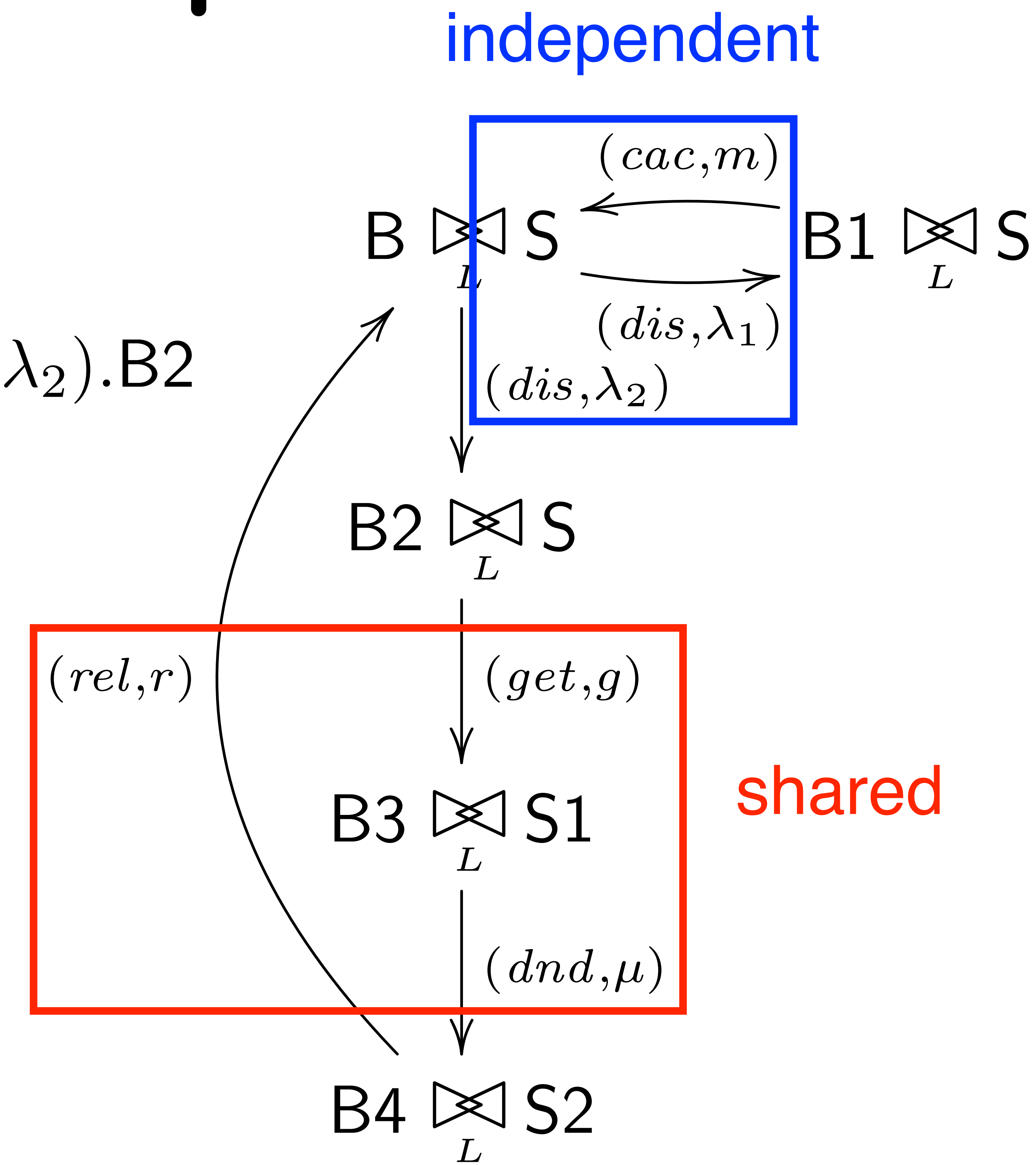
Example

$$\mathsf{S} \triangleq (get, \top).\mathsf{S1}$$
$$\mathsf{S1} \triangleq (dnd, \mu).\mathsf{S2}$$
$$\mathsf{S2} \triangleq (rel, \top).\mathsf{S}$$

$$\mathsf{B} \triangleq (dis, \lambda_1).\mathsf{B1} + (dis, \lambda_2).\mathsf{B2}$$
$$\mathsf{B1} \triangleq (cac, m).\mathsf{B}$$
$$\mathsf{B2} \triangleq (get, g).\mathsf{B3}$$
$$\mathsf{B3} \triangleq (dnd, \top).\mathsf{B4}$$
$$\mathsf{B4} \triangleq (rel, r).\mathsf{B}$$

$$L = \{get, dnd, rel\}$$

$$\mathsf{B} \underset{L}{\bowtie} \mathsf{S}$$



Example

$$\begin{aligned} \mathsf{S} &\triangleq (get, \top).\mathsf{S1} \\ \mathsf{S1} &\triangleq (dnd, \mu).\mathsf{S2} \\ \mathsf{S2} &\triangleq (rel, \top).\mathsf{S} \end{aligned}$$

$$L = \{get, dnd, rel\}$$

$$\begin{aligned} \mathsf{B} &\triangleq (dis, \lambda_1).\mathsf{B1} + (dis, \lambda_2).\mathsf{B2} \\ \mathsf{B1} &\triangleq (cac, m).\mathsf{B} \\ \mathsf{B2} &\triangleq (get, g).\mathsf{B3} \\ \mathsf{B3} &\triangleq (dnd, \top).\mathsf{B4} \\ \mathsf{B4} &\triangleq (rel, r).\mathsf{B} \end{aligned}$$

$$(\mathsf{B} \parallel \mathsf{B}) \underset{L}{\bowtie} \mathsf{S}$$

$$(\mathsf{B} \parallel \mathsf{B}) \underset{L}{\bowtie} \mathsf{S} \xrightarrow{(dis,\lambda_2)} (\mathsf{B2} \parallel \mathsf{B}) \underset{L}{\bowtie} \mathsf{S} \xrightarrow{(dis,\lambda_2)} (\mathsf{B2} \parallel \mathsf{B2}) \underset{L}{\bowtie} \mathsf{S}$$

$$\begin{array}{ccc} (\mathsf{B2} \parallel \mathsf{B2}) \underset{L}{\bowtie} \mathsf{S} & \xrightarrow{(get,g)} & (\mathsf{B3} \parallel \mathsf{B2}) \underset{L}{\bowtie} \mathsf{S1} \\ \Big\downarrow {\scriptstyle (get,g)} & & \\ (\mathsf{B2} \parallel \mathsf{B3}) \underset{L}{\bowtie} \mathsf{S1} & & \end{array}$$

$$r_{get}(\mathsf{B2}) = g$$

$$r_{get}(\mathsf{B2} \parallel \mathsf{B2}) = 2g$$

$$r_{get}(\mathsf{S}) = \top$$

$$r_{get}((\mathsf{B2} \parallel \mathsf{B2}) \underset{L}{\bowtie} \mathsf{S}) = 2g$$

Consumer/producer

$$
\begin{aligned}
Cons_1 &\stackrel{def}{=} (get, r_g).Cons_2 \\
Cons_2 &\stackrel{def}{=} (cons, r_c).Cons_1 \\
Prod_1 &\stackrel{def}{=} (make, r_m).Prod_2 \\
Prod_2 &\stackrel{def}{=} (put, r_p).Prod_1 \\
Buf_2 &\stackrel{def}{=} (get, \top).Buf_1 \\
Buf_1 &\stackrel{def}{=} (get, \top).Buf_0 \\
&\qquad + (put, \top).Buf_2 \\
Buf_0 &\stackrel{def}{=} (put, \top).Buf_1 \\
Sys &\stackrel{def}{=} Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_1
\end{aligned}
$$

Possible variants:

- A buffer with n places:

$$
\begin{aligned}
Buf_n &\stackrel{def}{=} (get, \top).Buf_{n-1} \\
Buf_i &\stackrel{def}{=} (get, \top).Buf_{i-1} \\
&\qquad + (put, \top).Buf_{i+1}, \\
&\qquad \text{for } 1 \leq i \leq n-1 \\
Buf_0 &\stackrel{def}{=} (put, \top).Buf_1
\end{aligned}
$$

- and k consumers:

$$
\overbrace{Cons_1 \parallel Cons_1 \parallel \ldots \parallel Cons_1}^{k} \underset{\{get\}}{\bowtie} Buf_n \underset{\{put\}}{\bowtie} Prod_1
$$

(taken from Mirco Tribastone's slides)

Consumer/producer

$$
\begin{array}{rclrcl}
Cons_1 & \stackrel{def}{=} & (get, r_g).Cons_2 & Prod_1 & \stackrel{def}{=} & (make, r_m).Prod_2 \\
Cons_2 & \stackrel{def}{=} & (cons, r_c).Cons_1 & Prod_2 & \stackrel{def}{=} & (put, r_p).Prod_1 \\
Buf_2 & \stackrel{def}{=} & (get, \top).Buf_1 & Buf_1 & \stackrel{def}{=} & (get, \top).Buf_0 + (put, \top).Buf_2 \\
Buf_0 & \stackrel{def}{=} & (put, \top).Buf_1 & Sys & \stackrel{def}{=} & Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_1
\end{array}
$$

$$
\dfrac{\dfrac{Cons_1 \xrightarrow{(get, r_g)} Cons_2 \qquad Buf_2 \xrightarrow{(get, \top)} Buf_1}{Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \xrightarrow{(get, r_g)} Cons_2 \underset{\{get\}}{\bowtie} Buf_1}}{Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_1 \xrightarrow{(get, r_g)} Cons_2 \underset{\{get\}}{\bowtie} Buf_1 \underset{\{put\}}{\bowtie} Prod_1}
$$

$$
Sys \xrightarrow{(get, r_g)} Cons_2 \underset{\{get\}}{\bowtie} Buf_1 \underset{\{put\}}{\bowtie} Prod_1
$$

(taken from Mirco Tribastone's slides)

Consumer/producer

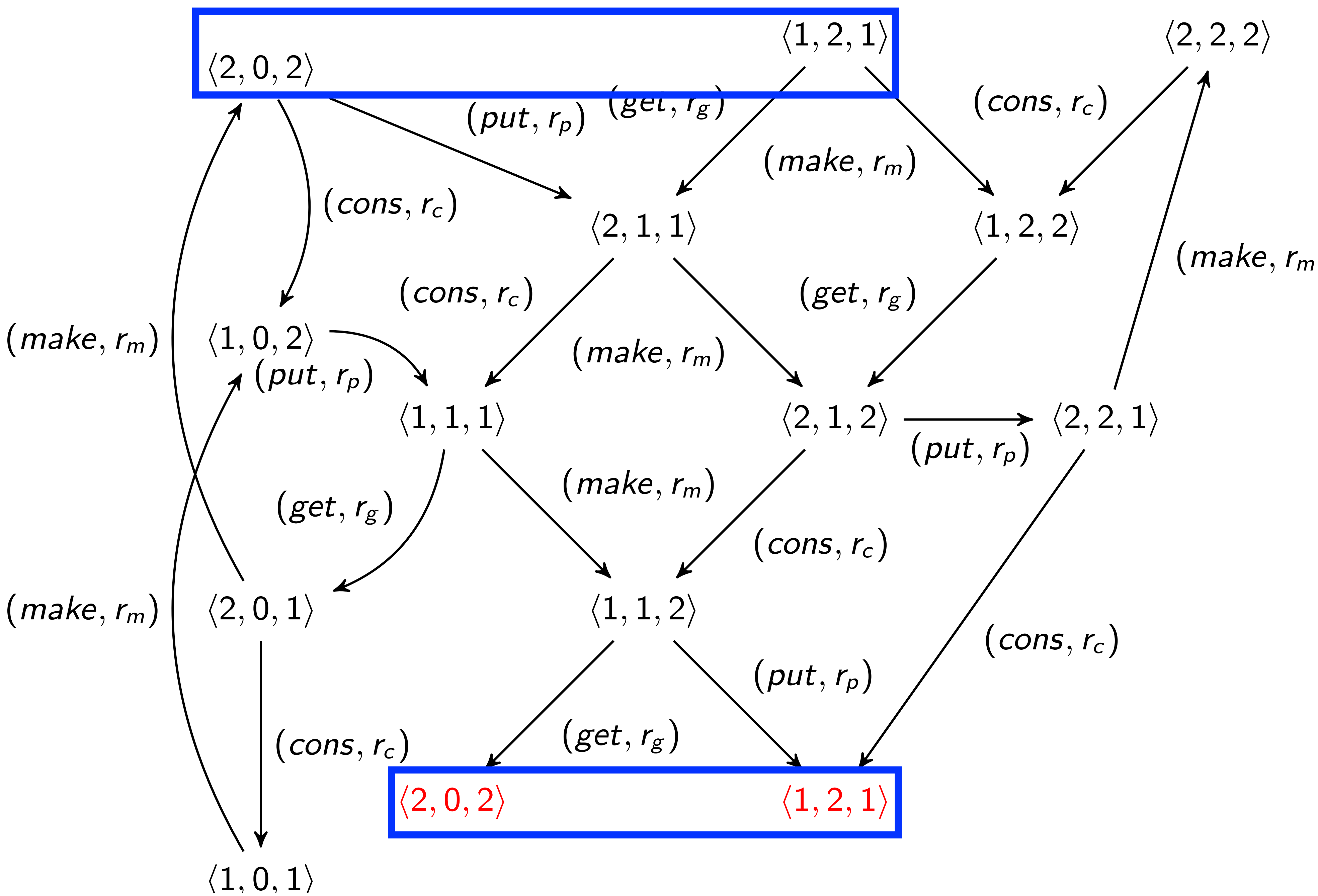
$$
\begin{array}{rcl@{\qquad}rcl}
Cons_1 & \stackrel{def}{=} & (get, r_g).Cons_2 & Prod_1 & \stackrel{def}{=} & (make, r_m).Prod_2 \\
Cons_2 & \stackrel{def}{=} & (cons, r_c).Cons_1 & Prod_2 & \stackrel{def}{=} & (put, r_p).Prod_1 \\
Buf_2 & \stackrel{def}{=} & (get, \top).Buf_1 & Buf_1 & \stackrel{def}{=} & (get, \top).Buf_0 + (put, \top).Buf_2 \\
Buf_0 & \stackrel{def}{=} & (put, \top).Buf_1 & Sys & \stackrel{def}{=} & Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_1
\end{array}
$$

$$
\frac{\dfrac{Prod_1 \xrightarrow{(make, r_m)} Prod_2}{Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{get\}}{\bowtie} Prod_1 \xrightarrow{(make, r_m)} Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_2}}{Sys \xrightarrow{(make, r_m)} Cons_1 \underset{\{get\}}{\bowtie} Buf_2 \underset{\{put\}}{\bowtie} Prod_2}
$$

(taken from Mirco Tribastone's slides)

Consumer/producer

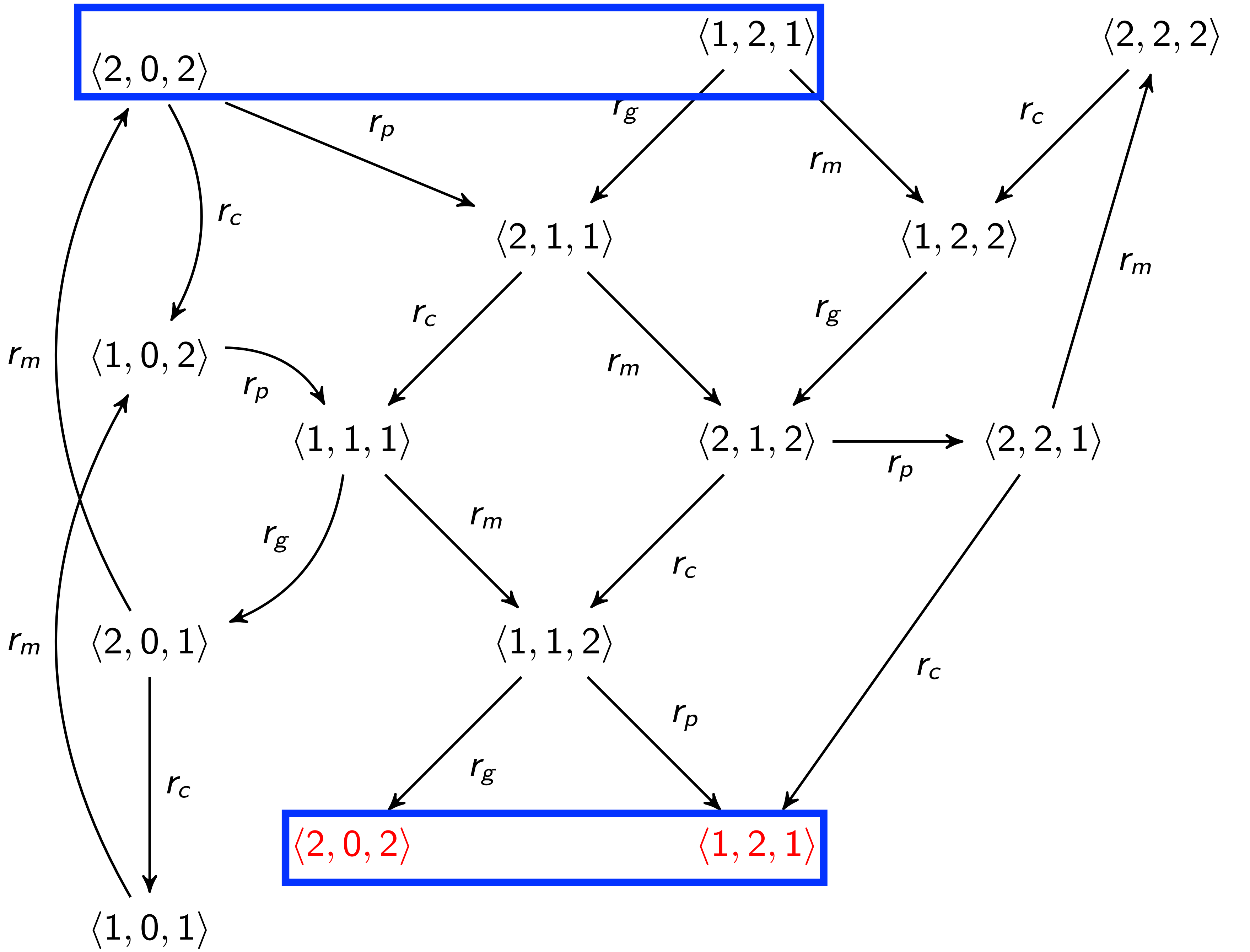
we may denote a state by $\langle i, j, k \rangle$ to indicate $Cons_i \underset{\{get\}}{\bowtie} Buf_j \underset{\{put\}}{\bowtie} Prod_k$



(taken from Mirco Tribastone's slides)

Consumer/producer

we may denote a state by $\langle i, j, k \rangle$ to indicate $Cons_i \underset{\{get\}}{\bowtie} Buf_j \underset{\{put\}}{\bowtie} Prod_k$



(taken from Mirco Tribastone's slides)

Bus maps

A third party app receives requests from users for live bus positioning information. It sends requests to the the Google Map API and the TfE Bus Info API and then aggregates the results to present a map view of the bus data which is returned to the user.

Construct a PEPA model to represent this system.

(taken from Jane Hillston's slides)

Bus maps

A third party app receives requests from users for live bus positioning information. It sends requests to the the Google Map API and the TfE Bus Info API and then aggregates the results to present a map view of the bus data which is returned to the user.

$$
\begin{aligned}
User &\stackrel{def}{=} (bus\_pos\_req, r).(bus\_pos\_resp, \top).User \\
Map\_finder &\stackrel{def}{=} (bus\_pos\_req, r).(google\_req, \lambda_1). \\
&\qquad (google\_resp, \top).(bus\_pos\_resp, \top).Map\_finder \\
Bus\_finder &\stackrel{def}{=} (bus\_pos\_req, r).(tfe\_req, \lambda_2). \\
&\qquad (tfe\_resp, \top).(bus\_pos\_resp, \top).Bus\_finder \\
Google &\stackrel{def}{=} (google\_req, \top).(google\_resp, \mu_1).Google \\
TfE &\stackrel{def}{=} (tfe\_req, \top).(tfe\_resp, \mu_2).Tfe \\
System &\stackrel{def}{=} User \underset{L}{\bowtie} \Big(Bus\_finder \underset{L}{\bowtie} Map\_finder) \underset{K}{\bowtie} (Google \parallel TfE)\Big)
\end{aligned}
$$

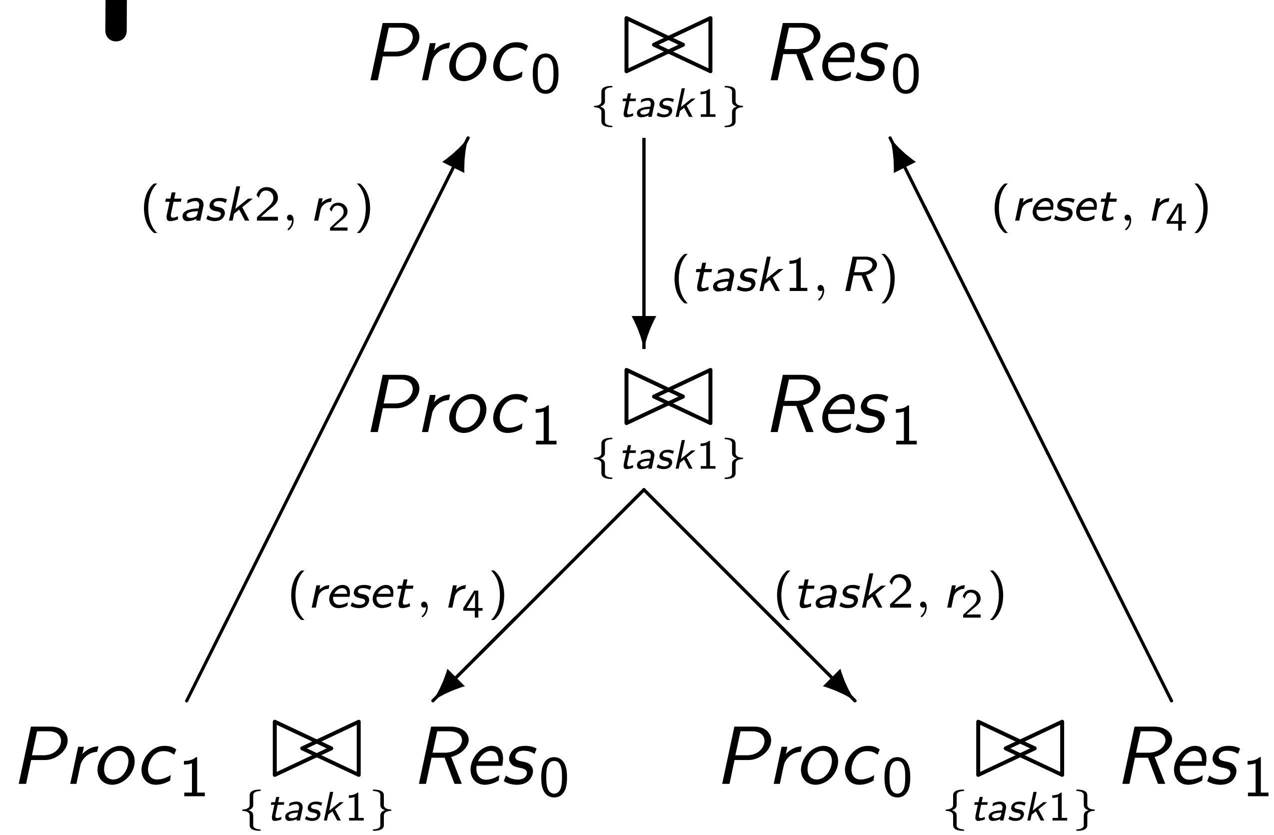
where $L = \{bus\_pos\_req, bus\_pos\_resp\}$ and
$K = \{google\_req, google\_resp, (tfe\_req, \top).(tfe\_resp, \mu_2)\}$.

(taken from Jane Hillston's slides)

Example

$$Proc_0 \stackrel{def}{=} (task1, r_1).Proc_1$$
$$Proc_1 \stackrel{def}{=} (task2, r_2).Proc_0$$
$$Res_0 \stackrel{def}{=} (task1, r_3).Res_1$$
$$Res_1 \stackrel{def}{=} (reset, r_4).Res_0$$

$$Proc_0 \underset{\{task1\}}{\bowtie} Res_0$$



$R = \min(r_1, r_3)$

$$\mathbf{Q} = \begin{pmatrix} -R & R & 0 & 0 \\ 0 & -(r_2 + r_4) & r_4 & r_2 \\ r_2 & 0 & -r_2 & 0 \\ r_4 & 0 & 0 & -r_4 \end{pmatrix}$$

$$\begin{cases} p \cdot Q = 0 \\ \sum_{i=1}^{N} p_i = 1 \end{cases}$$

(taken from Jane Hillston's slides)

Example

$$\mathbf{Q} = \begin{pmatrix} -R & R & 0 & 0 \\ 0 & -(r_2 + r_4) & r_4 & r_2 \\ r_2 & 0 & -r_2 & 0 \\ r_4 & 0 & 0 & -r_4 \end{pmatrix} \qquad \begin{cases} p \cdot Q = 0 \\ \displaystyle\sum_{i=1}^{N} p_i = 1 \end{cases}$$

$$r_1 = 2 \quad r_2 = 2 \quad r_3 = 6 \quad r_4 = 8 \quad R = \min\{r_1, r_3\} = 2$$

$$p_1 = \frac{20}{41} \quad p_2 = \frac{4}{41} \quad p_3 = \frac{1}{41} \quad p_4 = \frac{16}{41}$$

Reward structure

$\mathcal{C}$ a set of PEPA components

$\rho : \mathcal{C} \to \mathbb{R}$ a reward structure

p a steady state distribution

$$R_\rho \triangleq \sum_i p_i \cdot \rho(C_i)$$

sometimes rewards are defined in terms of activities

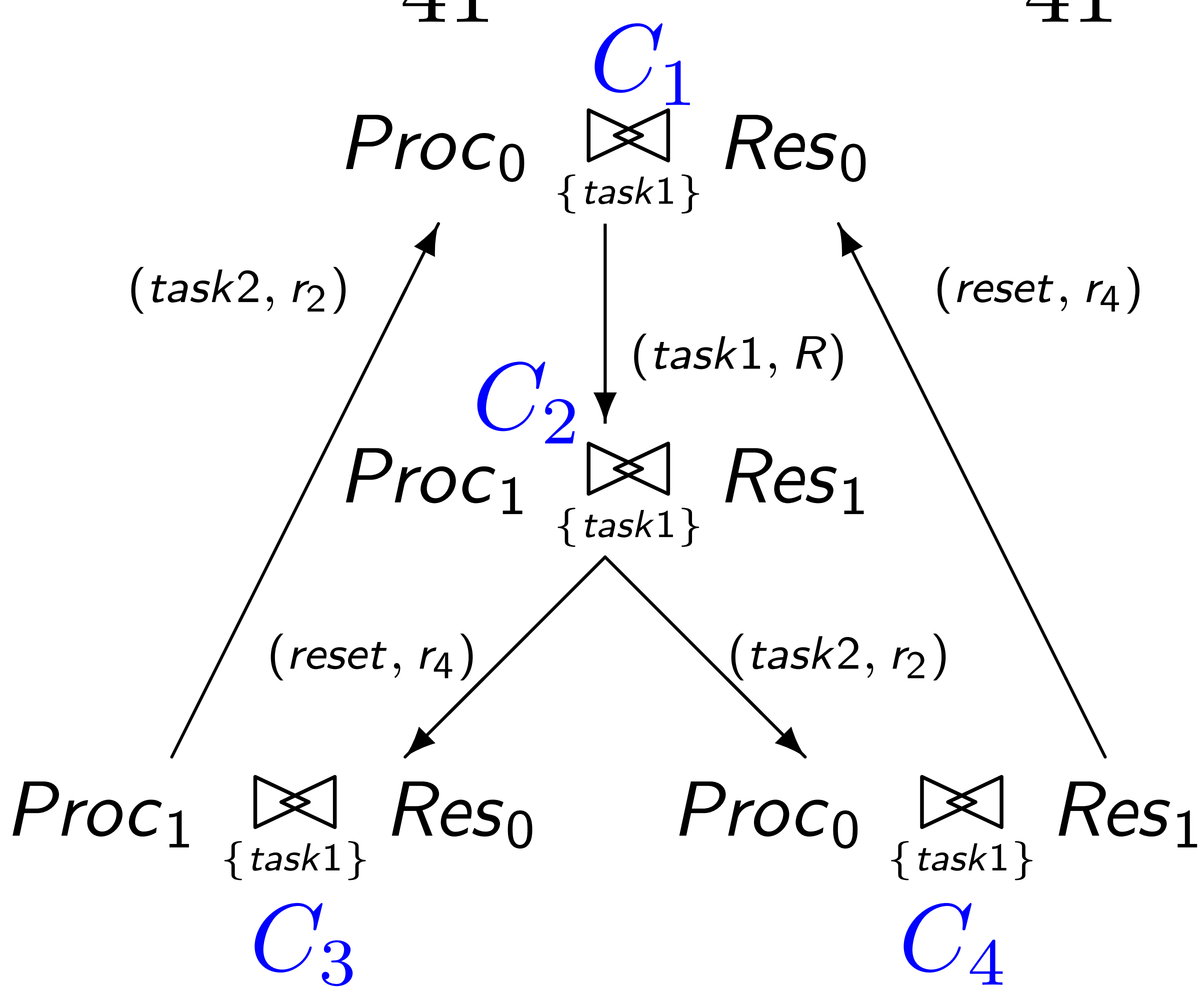
$$\rho : L \to \mathbb{R}$$

$$\rho(C) = \sum_{C \xrightarrow{(\alpha, r)} Q} \rho(\alpha)$$

Example: throughput

$$\mathbf{Q} = \begin{pmatrix} -R & R & 0 & 0 \\ 0 & -(r_2 + r_4) & r_4 & r_2 \\ r_2 & 0 & -r_2 & 0 \\ r_4 & 0 & 0 & -r_4 \end{pmatrix} \qquad \begin{cases} p \cdot Q = 0 \\ \sum_{i=1}^{N} p_i = 1 \end{cases}$$

$$p_1 = \frac{20}{41} \qquad p_2 = \frac{4}{41} \qquad p_3 = \frac{1}{41} \qquad p_4 = \frac{16}{41}$$



$$\rho(task_i) = 1 \qquad \rho(reset) = 0$$

$$\rho(C_1) = \rho(C_2) = \rho(C_3) = 1$$

$$\rho(C_4) = 0$$

$$R = \frac{20 + 4 + 1}{41} = \frac{25}{41} = 61\%$$

PEPA
further considerations

The importance of being Exp

$$\begin{tikzcd}
 & (\alpha, r).Stop \parallel (\beta, s).Stop \arrow[dl, "{(\alpha, r)}"'] \arrow[dr, "{(\beta, s)}"] & \\
Stop \parallel (\beta, s).Stop \arrow[dr, "{(\beta, s)}"'] & & (\alpha, r).Stop \parallel Stop \arrow[dl, "{(\alpha, r)}"] \\
 & Stop \parallel Stop &
\end{tikzcd}$$

We retain the expansion law of classical process algebra:

$$(\alpha, r).Stop \parallel (\beta, s).Stop =$$
$$(\alpha, r).(\beta, s).(Stop \parallel Stop) + (\beta, s).(\alpha, r).(Stop \parallel Stop)$$

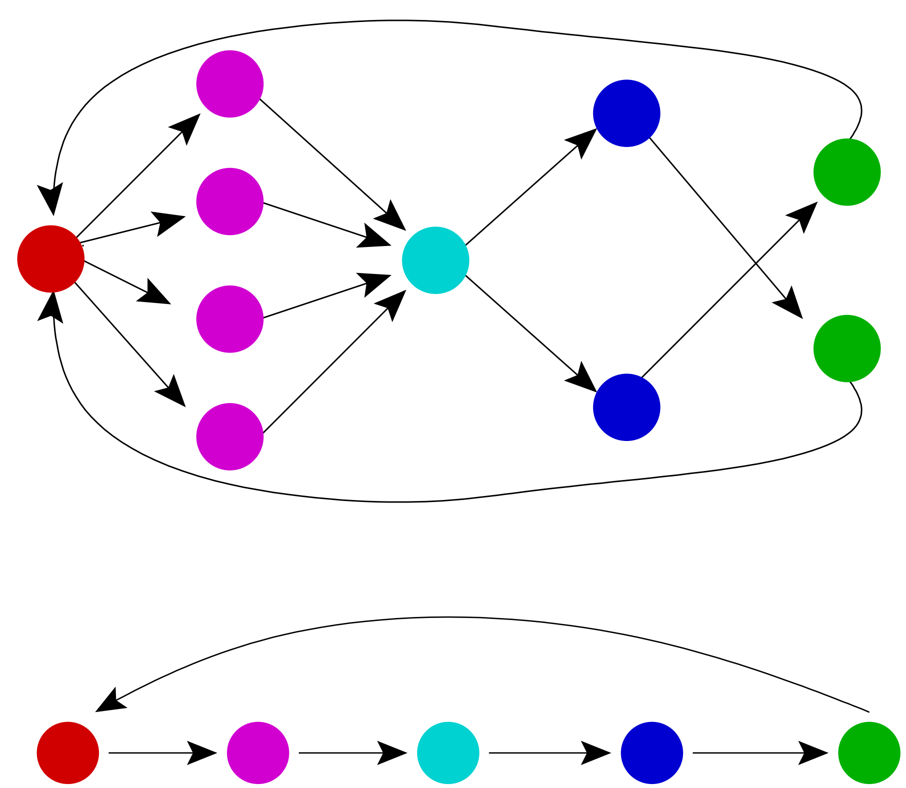
only if the negative exponential distribution is assumed.

(taken from Jane Hillston's slides)

Model aggregation

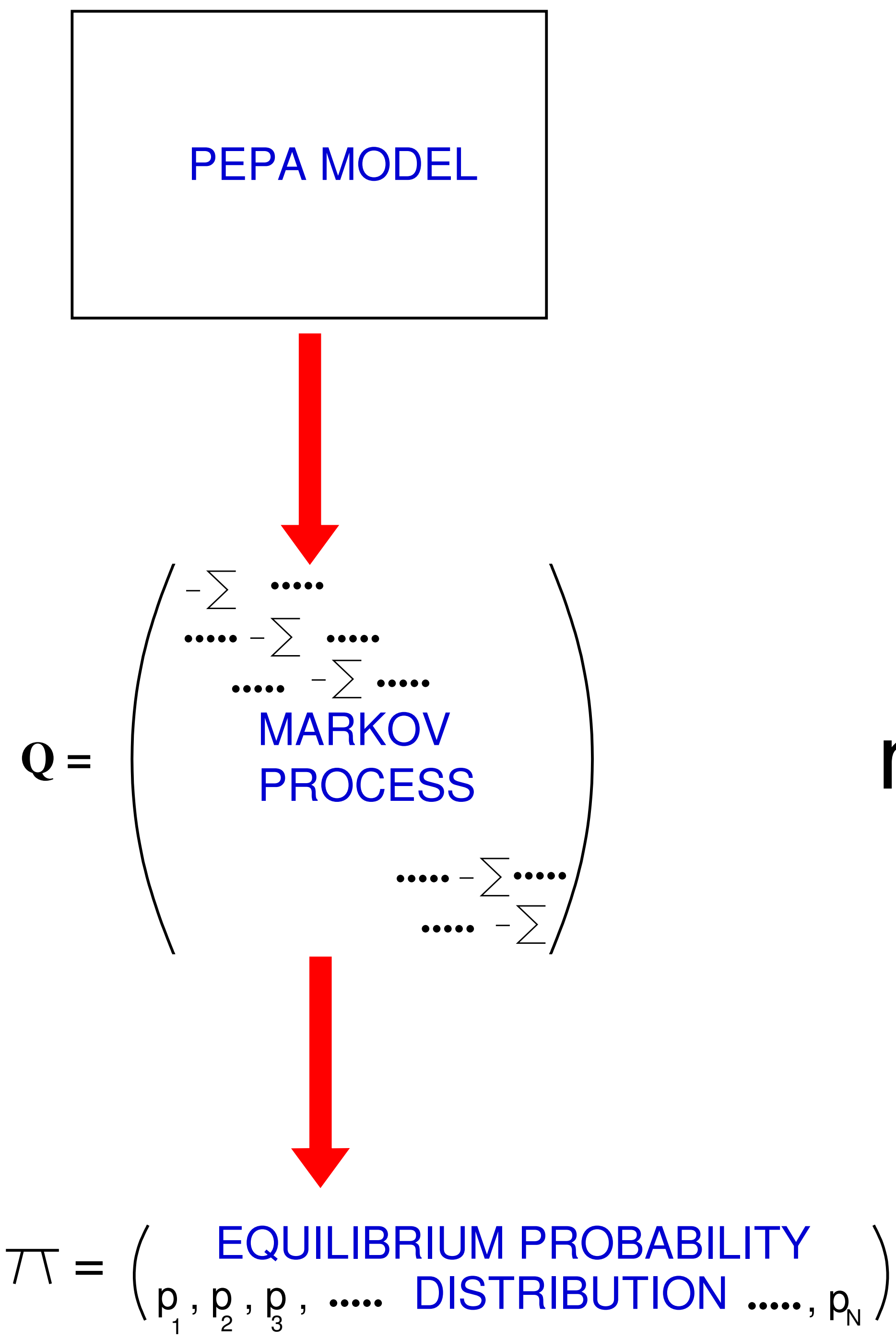
we can exploit CTMC bisimulation to reduce the state space (notion of lumpable partition)

it is the only equivalence that preserves the Markov property



(taken from Jane Hillston's slides)

Compositionality

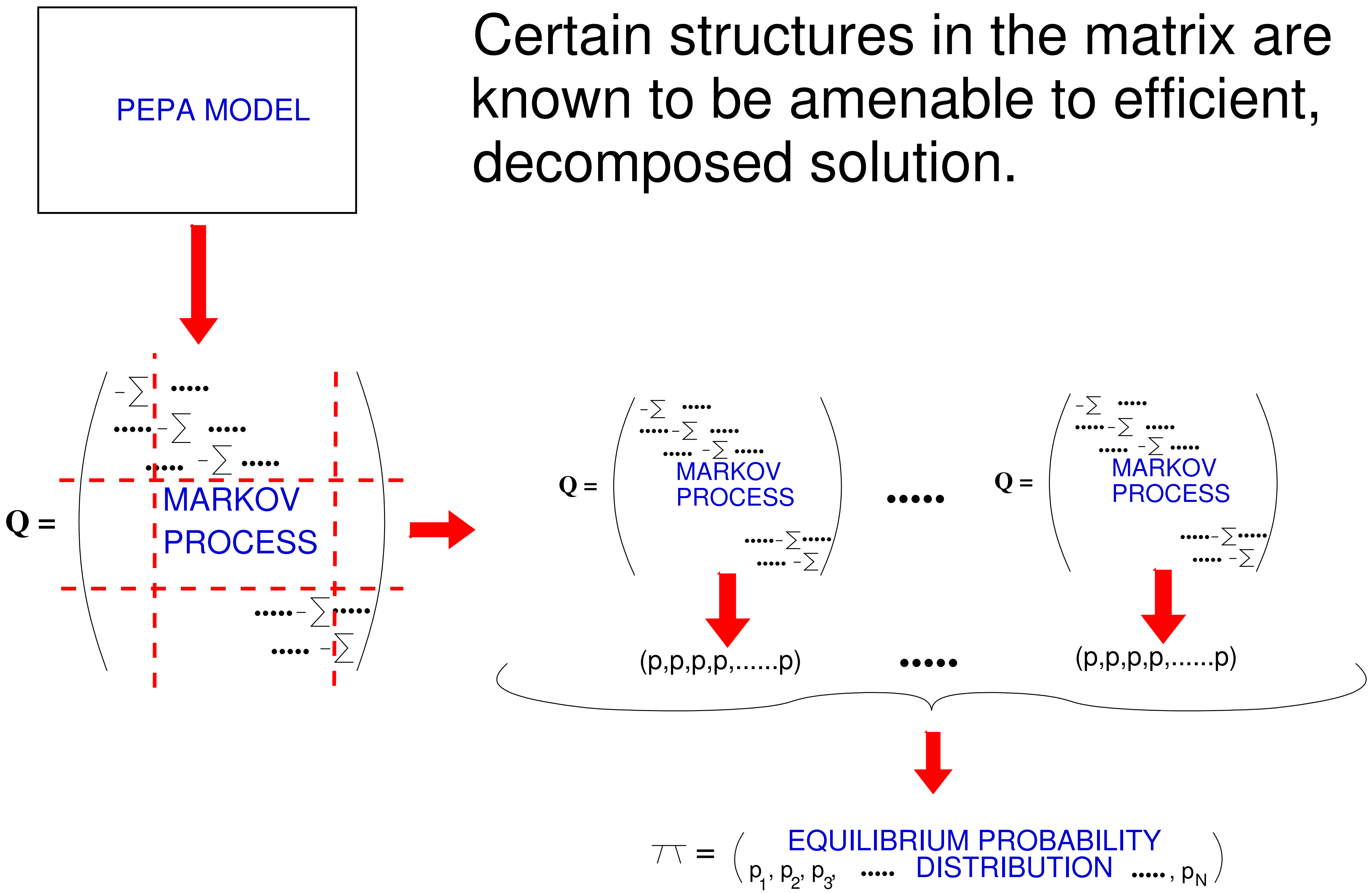


matrix size can grow very large

(taken from Jane Hillston's slides)

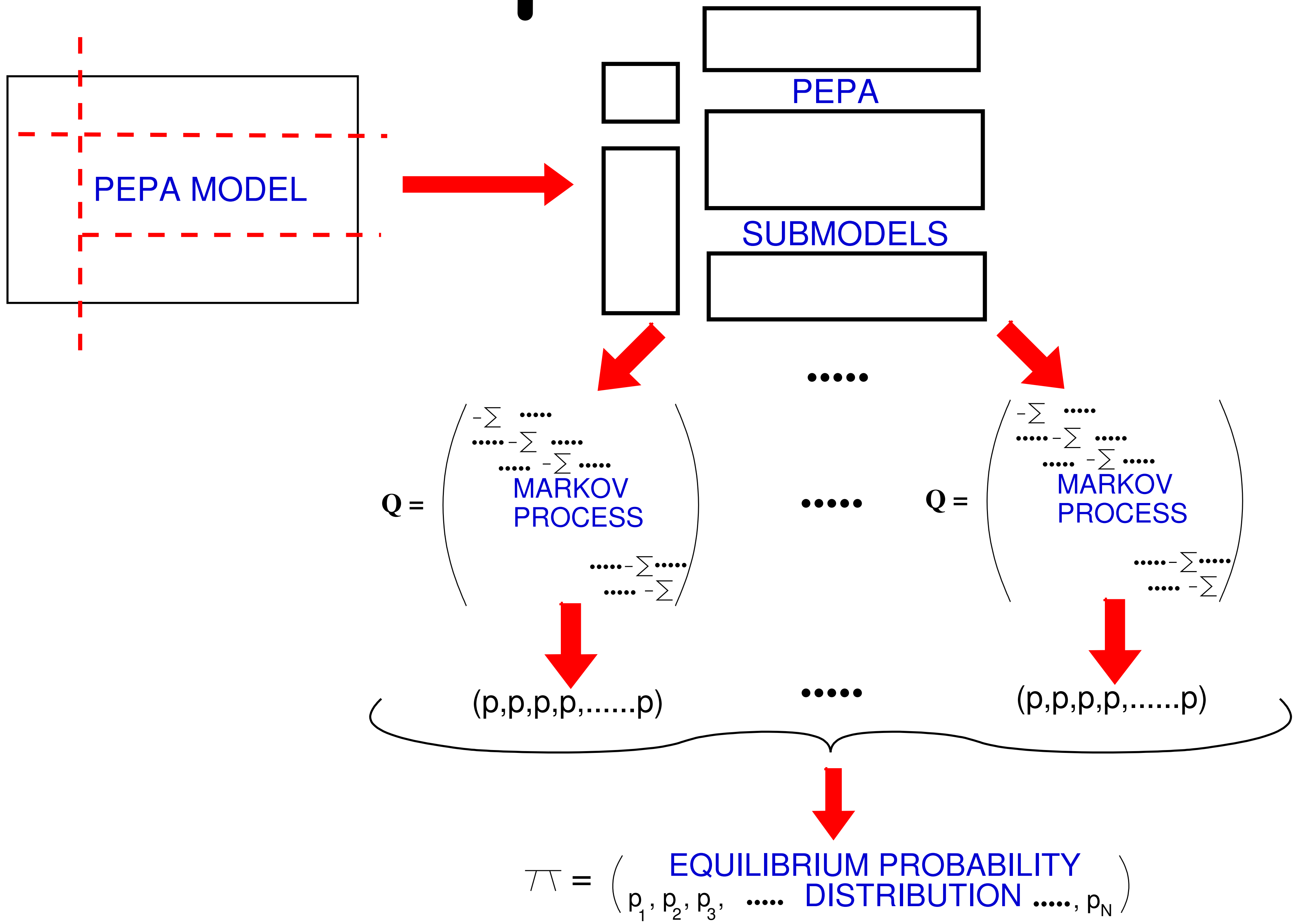
Compositionality

Certain structures in the matrix are known to be amenable to efficient, decomposed solution.



(taken from Jane Hillston's slides)

Compositionality



lift independent structures to the PEPA model!

(taken from Jane Hillston's slides)

Last badge

The final exam of a course consists of a list of 30 questions and a list of 30 answers.
Each student has to draw a bijective correspondence between the two lists, linking each question to its answer.
The teacher will assign 1 point to each correct link and 0 to each wrong link.

Unfortunately, many students had no time to prepare for the exam, because they had a tight deadline to deliver a project and they will answer completely random.

1. What is the average score for such students?

2. Would the average score be improved by adding more questions and answers?