FORMAL METHODS FOR SYSTEM VERIFICATION

Tackling state space explosion in PEPA models: strong equivalence

Sabina Rossi

DAIS
Università Ca' Foscari
Venezia

Equivalence between models

What is a notion of equivalence?

- An equivalence between models is a criterion which may be applied to determine whether two models can be considered to be, in some sense, indistinguishable.
- We are going to study a notion of bisimulation for PEPA models.

Bisimulation

Intuition

- Two agents are considered to be bisimilar when their externally observed behaviour appears to be the same.
- This is a formally defined notion of equivalence, based on the labelled transition system underlying the process algebra.
- These kind of equivalences form the basis of aggregation techniques for reducing the state space of the underlying Markov process, and thus provide a technique for making large models tractable.

Bisimulation for pure process algebra

Informal definition

- Bisimulation aims to capture the idea of equivalence as identical observed behaviour.
- If two agents are bisimilar it is not possible to distinguish between them by observation.
- We must specify which actions of the agents are considered visible to the observer.
- In its strongest form bisimilarity means that two agents are capable of exactly the same transitions, and the derivatives which result from the same transitions in the agents are themselves bisimilar.

Bisimulation for pure process algebra

Definition

- This notion of equivalence is based on the labelled transition system defined by the semantics of the language.
- Thus for a language whose labelled transition system is the triple $(\mathcal{P}, \mathcal{A}ct, \{\xrightarrow{\alpha} \mid \alpha \in \mathcal{A}ct\})$ the notion of bisimulation is expressed as follows.
- Two agents, $P, Q \in \mathcal{P}$, are strongly bisimilar, denoted $P \sim Q$, if and only if, there is some relation $\mathcal{R}$ over $\mathcal{P} \times \mathcal{P}$ such that if $(P, Q) \in \mathcal{R}$ then for all $\alpha \in \mathcal{A}ct$:
  - if $P \xrightarrow{\alpha} P'$, then for some Q', $Q \xrightarrow{\alpha} Q'$ and $(P', Q') \in \mathcal{R}$;
  - if $Q \xrightarrow{\alpha} Q'$, then for some P', $P \xrightarrow{\alpha} P'$ and $(P', Q') \in \mathcal{R}$.
- Thus, if P and Q are strongly bisimilar, any action performed by one must be matched by the other. Moreover, any subsequent action must also be matched.

Bisimulation for pure process algebra

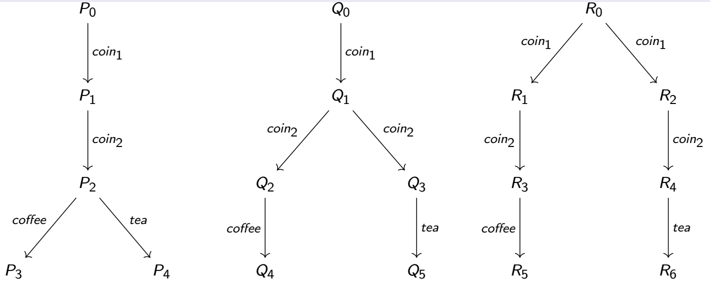
Example 1

- When defining a behavioral equivalence of concurrent systems described as LTSs, one might think that it is possible to consider systems equivalent if they give rise to the same (isomorphic) LTSs.
- Unfortunately, this would lead to unwanted distinctions, e.g., it would consider the two LTSs below different.
- Indeed, their behavior is the same: they can (only) execute infinitely many a-actions, and they should thus be considered equivalent.

$$\begin{tikzcd} P_0 \arrow[loop right, "a"] \end{tikzcd} \qquad\qquad \begin{tikzcd} Q_0 \arrow[d, "a"] \\ Q_1 \arrow[loop right, "a"] \end{tikzcd}$$

Example 2

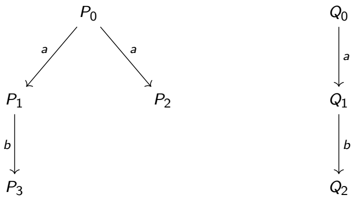
- Consider the following three systems representing the specification of three vending machines that accept two coins and deliver coffee or tea.
- The trace based equivalences equate all of them, the bisimulation based equivalences distinguish all of them.



Bisimulation for pure process algebra

Example 3

- Differently from trace-equivalence, bisimulation is sensitive to deadlocks.



Applications of equivalences between models

Entity-to-entity equivalences

- System-to-model: equivalences are used to establish the confidence in the model as a representation of the system being investigated.
- Model-to-model: equivalences are used to manipulate or compare models, in order to develop further knowledge about the system, or find alternative representations of the system.
- State-to-state: equivalences are used to simplify the model. When models are large and complex, model simplification strategies are required to reduce the complexity of the model. A set of equivalent states can be replaced by one macro-state.
- In the context of process algebra the concepts of state and model are interchangeable, both being represented as expressions in the language.

Strong equivalence

Intuition

- In PEPA two components are strongly bisimilar if
  - any a activity of one can be matched by an a activity of the other
  - every a-derivative of one is strongly bisimilar to some a-derivative of the the other
  - the apparent rates of all action types are the same in the two components.

Conditional transition rate

Definition

- The notion of strong equivalence is based on the notion of conditional transition rate.
- The conditional transition rate between two components P_i and P_j via a given action type α, is
$$q(P_i, P_j, \alpha) = \sum_{(\alpha, r_\alpha) \in \mathcal{A}ct(P_i|P_j)} r_\alpha$$
  where $\mathcal{A}ct(P_i|P_j) = \{\!|(\alpha, r_\alpha) \in \mathcal{A}ct(P_i)|\ P_i \xrightarrow{(\alpha, r_\alpha)} P_j|\!\}$.
- This is the rate at which a system behaving as component P_i evolves to behave as component P_j as a result of completing an activity of action type α.
- It is the sum of activity rates, labelling arcs of type α, connecting the nodes corresponding to P_i and P_j in the derivation graph.

Total conditional transition rate

Definition

- Let S be a set of possible derivatives. The total conditional transition rate from P_i to S, denoted $q[P_i, S, \alpha]$ is defined by:

$$q[P_i, S, \alpha] = \sum_{P_j \in S} q(P_i, P_j, \alpha).$$

- Two PEPA components are strongly equivalent if there is an equivalence relation between them such that, for any action type α, the total conditional transition rates from those components to any equivalence class, via activities of type α are the same.

Strong equivalence

Definition

- An equivalence relation $\mathcal{R} \subseteq \mathcal{C} \times \mathcal{C}$ is a strong equivalence if whenever $(P, Q) \in \mathcal{R}$ then for all $\alpha \in \mathcal{A}$ and for all $S \in \mathcal{C}/\mathcal{R}$,

$$q[P, S, \alpha] = q[Q, S, \alpha].$$

- The identity relation is a strong equivalence.

Transitive closure of a union of relations

Definition

- Let $\mathcal{R}_i$ with $i \in I$ for some index set I be a family of strong equivlences.
- The transitive closure of their union, denoted by $\mathcal{R} = (\cup_{i \in I} \mathcal{R}_i)^*$, is defined as follows.

$$\mathcal{R} = lim_{n \to \infty} \mathcal{R}^n$$

where

- $\mathcal{R}^0 = (\cup_{i \in I} \mathcal{R}_i)$
- $\mathcal{R}^n = (\cup_{i \in I} \mathcal{R}_i); \mathcal{R}^{n-1}$

with ; denoting the composition of two relations, i.e., $(P, Q) \in \mathcal{R}_1; \mathcal{R}_2$ if there exists R such that $(P, R) \in \mathcal{R}_1$ and $(R, Q) \in \mathcal{R}_2$.

The largest strong equivalence

Theorem

- We are interested in the largest strong equivalence, formed by the union of all strong equivalences.
- Let $\mathcal{R}_i$ with $i \in I$ for some index set I be a family of strong equivlences.
- Then $\mathcal{R} = (\cup_{i \in I} \mathcal{R}_i)^*$, the transitive closure of their union, is also a strong equivalence.

Strongly equivalent PEPA components

Definition

- Two PEPA components P and Q are said to be strongly equivalent, written $P \cong Q$, if $(P, Q) \in \mathcal{R}$ for some strong equivalence $\mathcal{R}$, i.e.,

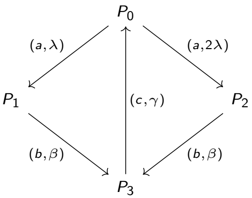
$$\cong = \cup\{\mathcal{R} | \, \mathcal{R} \text{ is a strong equivalence}\}.$$

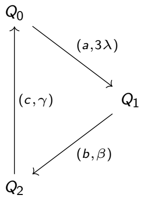
- $\cong$ is the largest strong equivalence.
- In order to show that $P \cong Q$ we must find a strong equivalence relation $\mathcal{R}$ such that $(P, Q) \in \mathcal{R}$.

Example 4

- Consider the following two PEPA models:

$$\begin{aligned} P_0 &\stackrel{\text{def}}{=} (a,\lambda).P_1 + (a,2\lambda).P_2 \\ P_1 &\stackrel{\text{def}}{=} (b,\beta).P_3 \\ P_2 &\stackrel{\text{def}}{=} (b,\beta).P_3 \\ P_3 &\stackrel{\text{def}}{=} (c,\gamma).P_0 \end{aligned} \qquad \begin{aligned} Q_0 &\stackrel{\text{def}}{=} (a,3\lambda).Q_1 \\ Q_1 &\stackrel{\text{def}}{=} (b,\beta).Q_2 \\ Q_2 &\stackrel{\text{def}}{=} (c,\gamma).Q_0 \end{aligned}$$





Strongly equivalent PEPA components

Example 4

- We prove that P_0 and Q_0 are strongly equivalent.
- Indeed, consider the reflexive, symmetric and transitive closure of the following relation:

$$\mathcal{R} = \{(P_0, Q_0), (P_1, Q_1), (P_2, Q_1), (P_3, Q_2)\}$$

- $\mathcal{R}$ induces the following equivalence classes:

$$C_1 = \{P_0, Q_0\} \quad C_2 = \{P_1, P_2, Q_1\} \quad C_3 = \{P_3, Q_2\}$$

Strongly equivalent PEPA components

Example 4: We prove that $P_0 \cong Q_0$.

- $q[P_0, C_1, x] = q[Q_0, C_1, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_0, C_2, x] = q[Q_0, C_2, x] = 0$ for $x \in \{b, c\}$
- $q[P_0, C_3, x] = q[Q_0, C_3, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_0, C_2, a] = q(P_0, P_1) + q(P_0, P_2) = \lambda + 2\lambda = 3\lambda$
  $q[Q_0, C_2, a] = q(Q_0, Q_1) = 3\lambda$

Strongly equivalent PEPA components

Example 4: We prove that $P_0 \cong Q_0$.

- $q[P_1, C_1, x] = q[P_2, C_1, x] = q[Q_1, C_1, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_1, C_2, x] = q[P_2, C_2, x] = q[Q_1, C_2, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_1, C_3, x] = q[P_2, C_3, x] = q[Q_1, C_3, x] = 0$ for $x \in \{a, c\}$
- $q[P_1, C_3, b] = q(P_1, P_3) = \beta$
  $q[P_2, C_3, b] = q(P_2, P_3) = \beta$
  $q[Q_1, C_3, b] = q(Q_1, Q_2) = \beta$

Strongly equivalent PEPA components

Example 4: We prove that $P_0 \cong Q_0$.

- $q[P_3, C_1, x] = q[Q_2, C_1, x] = 0$ for $x \in \{a, b\}$
- $q[P_3, C_2, x] = q[Q_2, C_2, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_3, C_3, x] = q[Q_2, C_3, x] = 0$ for $x \in \{a, b, c\}$
- $q[P_3, C_1, c] = q(P_3, P_0) = \gamma$
  $q[Q_2, C_1, c] = q(Q_2, Q_0) = \gamma$

Properties of strong equivalence

Congruence

- Strong equivalence is a congruence, i.e., it is preserved by all the combinators of the language.
- Preservation by combinators: if $P_1 \cong P_2$ then
  - $a.P_1 \cong a.P_2$
  - $P_1 + Q \cong P_2 + Q$
  - $P_1 \underset{L}{\bowtie} Q \cong P_2 \underset{L}{\bowtie} Q$
  - $P_1/L \cong P_2/L$

Properties of strong equivalence

Preservation by recursive definitions.

- Let E and F be two PEPA component expressions containing variables $\tilde{X}$ at most. Then $E \cong F$ if, for all indexed sets of components $\tilde{P}$, $E\{\tilde{P}/\tilde{X}\} \cong F\{\tilde{P}/\tilde{X}\}$.
- Replacing a subexpression by a strongly equivalent subexpression, will result in a component expression which is strongly equivalent to the original.
- Let $\tilde{E}$ and $\tilde{F}$ contain variable $\tilde{X}$ at most. Let $\tilde{A} \stackrel{\mathrm{def}}{=} \tilde{E}\{\tilde{A}/\tilde{X}\}$, $\tilde{B} \stackrel{\mathrm{def}}{=} \tilde{F}\{\tilde{B}/\tilde{X}\}$ and $\tilde{E} \cong \tilde{F}$. Then $\tilde{A} \cong \tilde{B}$.

Properties of strong equivalence

Equational laws - (Choice)

- $P + Q \cong Q + P$
- $P + (Q + R) \cong (P + Q) + R$

Equational laws - (Constant)

- If $A \stackrel{\text{def}}{=} P$ then $A \cong P$

Properties of strong equivalence

Equational laws - (Hiding)

- $(P + Q)/L \cong P/L + Q/L$
- $((\alpha, r).P)/L \cong \begin{cases} (\tau, r).P/L & \text{if } \alpha \in L \\ (\alpha, r).P/L & \text{if } \alpha \notin L \end{cases}$
- $(P/L)/K \cong P/(L \cup K)$
- $P/L \cong P \quad \text{if } L \cap \vec{\mathcal{A}}(P) = \emptyset \quad \text{where } \vec{\mathcal{A}}(P) = \bigcup_{P_i \in ds(P)} \mathcal{A}(P_i)\,.$

Properties of strong equivalence

Equational laws - (Cooperation)

- $P \underset{L}{\bowtie} Q \cong Q \underset{L}{\bowtie} P$
- $P \underset{L}{\bowtie} (Q \underset{L}{\bowtie} R) \cong (P \underset{L}{\bowtie} Q) \underset{L}{\bowtie} R$
- $(P \underset{L}{\bowtie} Q)/K \cong (P/K) \underset{L}{\bowtie} (Q/K) \quad \text{where } K \cap L = \emptyset$
- $P \underset{K}{\bowtie} Q \cong P \underset{L}{\bowtie} Q \quad \text{if } K \cap \left(\vec{\mathcal{A}}(P) \cup \vec{\mathcal{A}}(Q)\right) = L$
- $(P \underset{L}{\bowtie} Q) \underset{K}{\bowtie} R \cong$
$$\begin{cases} P \underset{L}{\bowtie} (Q \underset{K}{\bowtie} R) & \text{if } \vec{\mathcal{A}}(R) \cap L \setminus K = \emptyset \wedge \vec{\mathcal{A}}(P) \cap K \setminus L = \emptyset \\ Q \underset{L}{\bowtie} (P \underset{K}{\bowtie} R) & \text{if } \vec{\mathcal{A}}(R) \cap L \setminus K = \emptyset \wedge \vec{\mathcal{A}}(P) \cap K \setminus L = \emptyset \end{cases}$$

Properties of strong equivalence

Equational laws - (Expansion law)

- Let $P \equiv (P_1 \underset{L}{\bowtie} P_2)/K$. Then

$$
\begin{aligned}
P \quad \cong \quad & \sum\{(\alpha, r).(P_1' \underset{L}{\bowtie} P_2)/K \mid P_1 \xrightarrow{(\alpha,r)} P_1',\, \alpha \notin L \cup K\} \\
+ \quad & \sum\{(\alpha, r).(P_1 \underset{L}{\bowtie} P_2')/K \mid P_2 \xrightarrow{(\alpha,r)} P_2',\, \alpha \notin L \cup K\} \\
+ \quad & \sum\{(\tau, r).(P_1' \underset{L}{\bowtie} P_2)/K \mid P_1 \xrightarrow{(\alpha,r)} P_1',\, \alpha \in K \setminus L\} \\
+ \quad & \sum\{(\tau, r).(P_1 \underset{L}{\bowtie} P_2')/K \mid P_2 \xrightarrow{(\alpha,r)} P_2',\, \alpha \in K \setminus L\} \\
+ \quad & \sum\{(\alpha, r).(P_1' \underset{L}{\bowtie} P_2')/K \mid P_1 \xrightarrow{(\alpha,r_1)} P_1',\, P_2 \xrightarrow{(\alpha,r_2)} P_2', \\
& \alpha \in L \setminus K,\, r = \frac{r_1}{r_\alpha(P_1)} \frac{r_2}{r_\alpha(P_2)} min(r_\alpha(P_1), r_\alpha(P_2))\} \\
+ \quad & \sum\{(\tau, r).(P_1' \underset{L}{\bowtie} P_2')/K \mid P_1 \xrightarrow{(\alpha,r_1)} P_1',\, P_2 \xrightarrow{(\alpha,r_2)} P_2', \\
& \alpha \in L \cap K,\, r = \frac{r_1}{r_\alpha(P_1)} \frac{r_2}{r_\alpha(P_2)} min(r_\alpha(P_1), r_\alpha(P_2))\}
\end{aligned}
$$

Strong equivalence and system components

Implications for system components

- Let Sys_P and Sys_Q denote the system components modelled by P and Q, respectively.
- If $P \cong Q$ then
  - $\mathcal{A}(P) = \mathcal{A}(Q)$
  - $r_\alpha(P) = r_\alpha(Q)$ for all $\alpha \in \mathcal{A}$
  - $q(P) = q(Q)$
- Hence Sys_P and Sys_Q appear to perform the same actions, at the same rates, and their expected delay before performing some action will be the same.
- Thus an external observer would be unable to distinguish between them on the basis of a memoryless observation.

Strong equivalence and the Markov process

Implications for the Markov process

- The relation $\cong$ restricted to the derivative set of any component P partitions this set.
- Let $ds(P)/\cong$ denote the set of equivalence classes generated in this way.
- For any component P, $ds(P)/\cong$ induces a strong lumpability on the state space of the Markov process corresponding to P.

Strong equivalence and aggegation

Applications of strong equivalence

- Each equivalence class $S \in ds(P)/\cong$ represents a set of derivatives which all exhibit the same behaviour.
- This corresponds to a lumpable partition within the state space of the Markov process.
- We can thus construct the aggregated Markov chain having a state corresponding to each of the equivalence classes induced on the derivative set by strong equivalence.

Example 5

- Consider the following two PEPA models:

$$
\begin{aligned}
P_0 &\stackrel{\text{def}}{=} (a,\lambda).P_1 + (a,2\lambda).P_2 \\
P_1 &\stackrel{\text{def}}{=} (b,\beta).P_3 \\
P_2 &\stackrel{\text{def}}{=} (b,\beta).P_3 \\
P_3 &\stackrel{\text{def}}{=} (c,\gamma).P_0
\end{aligned}
\qquad
\begin{aligned}
Q_0 &\stackrel{\text{def}}{=} (a,3\lambda).Q_1 \\
Q_1 &\stackrel{\text{def}}{=} (b,\beta).Q_2 \\
Q_2 &\stackrel{\text{def}}{=} (c,\gamma).Q_0
\end{aligned}
$$

$$
\begin{aligned}
R &\stackrel{\text{def}}{=} (d,\delta).(m,\mu).R \\
S &\stackrel{\text{def}}{=} (a,\alpha).P_0 + R \\
T &\stackrel{\text{def}}{=} (a,\alpha).Q_0 + R
\end{aligned}
$$

- Are S and T strongly equivalent?

Example 5

- We have already proved that $P_0 \cong Q_0$. Moreover,
- $(a, \alpha).P_0 \cong (a, \alpha).Q_0$ (preservation by prefix)
- $(a, \alpha).P_0 + R \cong (a, \alpha).Q_0 + R$ (preservation by sum)
- We can conclude that $S \cong T$
- No extra reasoning on S and T is needed.

Compositional reasoning

Example 6

- Consider the following two PEPA models:

$$
\begin{aligned}
A &\stackrel{\text{def}}{=} (P \| Q)/L \quad \text{with } L = \{\beta, \delta\} \\
B &\stackrel{\text{def}}{=} (P/L) \| (Q/L) \quad \text{with } L = \{\beta, \delta\} \\
S' &\stackrel{\text{def}}{=} A \underset{\{\alpha,\gamma\}}{\bowtie} R \\
S'' &\stackrel{\text{def}}{=} B \underset{\{\alpha,\gamma\}}{\bowtie} R
\end{aligned}
$$

- Are S and T strongly equivalent?

Example 6

- We prove that $A \cong B$, i.e., $(P \| Q)/L \cong (P/L) \| (Q/L)$. Indeed,

$$\begin{array}{rcll} (P\|Q)/L & = & (P \underset{\emptyset}{\bowtie} Q)/L & \\ & \cong & (P/L) \underset{\emptyset}{\bowtie} (Q/L) & \text{since } L \cap \emptyset = \emptyset \\ & \cong & (P/L)\|(Q/L) & \end{array}$$

- Now, from preservation by cooperation, since $A \cong B$ we have

$$A \underset{\{\alpha,\gamma\}}{\bowtie} R \cong B \underset{\{\alpha,\gamma\}}{\bowtie} R$$

i.e.,

$$S' \cong S''.$$

Example 7

- Consider the following PEPA model:

$$
\begin{aligned}
P &\stackrel{\text{def}}{=} (\alpha, \top).(\beta, r).P \\
Q &\stackrel{\text{def}}{=} (\alpha, s).(\gamma, t).Q \\
S &\stackrel{\text{def}}{=} (P \| P) \underset{\{\alpha\}}{\bowtie} Q
\end{aligned}
$$

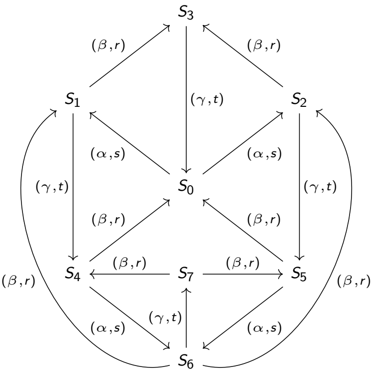
Example 7

- Draw the derivation graph of S:

$$
\begin{aligned}
S_0 &\stackrel{\text{def}}{=} (P \| P) \underset{\{\alpha\}}{\bowtie} Q \\
S_1 &\stackrel{\text{def}}{=} ((\beta, r).P \| P) \underset{\{\alpha\}}{\bowtie} (\gamma, t).Q \\
S_2 &\stackrel{\text{def}}{=} (P \| (\beta, r).P) \underset{\{\alpha\}}{\bowtie} (\gamma, t).Q \\
S_3 &\stackrel{\text{def}}{=} (P \| P) \underset{\alpha}{\bowtie} (\gamma, t).Q \\
S_4 &\stackrel{\text{def}}{=} ((\beta, r).P \| P) \underset{\{\alpha\}}{\bowtie} Q \\
S_5 &\stackrel{\text{def}}{=} (P \| (\beta, r).P) \underset{\{\alpha\}}{\bowtie} Q \\
S_6 &\stackrel{\text{def}}{=} ((\beta, r).P \| (\beta, r).P) \underset{\{\alpha\}}{\bowtie} (\gamma, t).Q \\
S_7 &\stackrel{\text{def}}{=} ((\beta, r).P \| (\beta, r).P) \underset{\{\alpha\}}{\bowtie} Q
\end{aligned}
$$

Example 7

- The derivation graph of S is:



Example 7

- Consider the systems:

$$
\begin{aligned}
S' &\stackrel{\text{def}}{=} ((P \| P) \underset{\alpha}{\bowtie} Q)/\{\gamma\} \\
S'' &\stackrel{\text{def}}{=} (P \| P) \underset{\alpha}{\bowtie} (Q/\{\gamma\})
\end{aligned}
$$

- Are S' and S'' strongly equivalent? Indeed,

$$
\begin{aligned}
S' &= ((P \| P) \underset{\alpha}{\bowtie} Q)/\{\gamma\} \\
&\cong ((P \| P)/\{\gamma\}) \underset{\alpha}{\bowtie} (Q/\{\gamma\}) && \text{since } \{\alpha\} \cap \{\gamma\} = \emptyset \\
&\cong (P \| P) \underset{\alpha}{\bowtie} (Q/\{\gamma\}) && \text{since } \{\gamma\} \cap \vec{\mathcal{A}}(P \| P) = \emptyset \\
&= S''.
\end{aligned}
$$

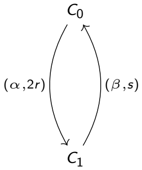
- We have proved that $S' \cong S''$.

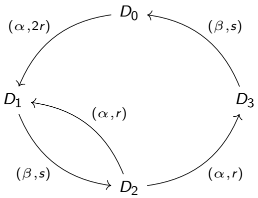
Strongly equivalent PEPA components

Example 8

- Consider the following two PEPA models:

$$
\begin{aligned}
C_0 &\stackrel{\text{def}}{=} (\alpha, 2r).C_1 \\
C_1 &\stackrel{\text{def}}{=} (\beta, s).C_0
\end{aligned}
\qquad
\begin{aligned}
D_0 &\stackrel{\text{def}}{=} (\alpha, 2r).D_1 \\
D_1 &\stackrel{\text{def}}{=} (\beta, s).D_2 \\
D_2 &\stackrel{\text{def}}{=} (\alpha, r).D_3 + (\alpha, r).D_1 \\
D_3 &\stackrel{\text{def}}{=} (\beta, s).D_0
\end{aligned}
$$





Strongly equivalent PEPA components

Example 8

- We prove that C_0 and D_0 are strongly equivalent.
- Indeed, consider the reflexive, symmetric and transitive closure of the following relation:

$$\mathcal{R} = \{(C_0, D_0), (C_0, D_2), (C_1, D_1), (C_1, D_3)\}$$

- $\mathcal{R}$ induces the following equivalence classes:

$$S_1 = \{C_0, D_0, D_2\} \quad S_2 = \{C_1, D_1, D_3\}$$

Strongly equivalent PEPA components

Example 8: We prove that $C_0 \cong D_0$.

- $q[C_0, S_1, x] = q[D_0, S_1, x] = q[D_2, S_1, x] = 0$ for $x \in \{\alpha, \beta\}$
- $q[C_0, S_2, \beta] = q[D_0, S_2, \beta] = q[D_2, S_2, \beta] = 0$
- $q[C_0, S_2, \alpha] = q(C_0, C_1, \alpha) = 2r$
- $q[D_0, S_2, \alpha] = q(D_0, D_1, \alpha) = 2r$
- $q[D_2, S_2, \alpha] = q(D_2, D_1, \alpha) + q(D_2, D_3, \alpha) = r + r = 2r$

- $q[C_1, S_2, x] = q[D_1, S_2, x] = q[D_3, S_2, x] = 0$ for $x \in \{\alpha, \beta\}$
- $q[C_1, S_1, \alpha] = q[D_1, S_1, \alpha] = q[D_3, S_1, \alpha] = 0$
- $q[C_1, S_1, \beta] = q[D_1, S_1, \beta] = q[D_3, S_1, \beta] = s$

Strongly equivalent PEPA components

Example 8: lumpable partition

- Let us restrict $\cong$ to the derivative set of D_0, $ds(D_0)$.
- The set $ds(D_0)/\cong$ contains two equivalence classes:

$$[D_0] = \{D_0, D_2\} \qquad [D_1] = \{D_1, D_3\}$$

- $\cong$ induces a partition of $ds(D_0)$ which is a strong lumpability.
- The aggegated Markov chain is:

$$\begin{tikzcd} {[D_0]} \arrow[d, bend right, "(\alpha,2r)"] \\ {[D_1]} \arrow[u, bend right, "(\beta,s)"'] \end{tikzcd}$$

Strong equivalence and lumpability

Example 9

- Consider a system consisting of the following components: *Comp*, *Res* and *Repman*.
- *Comp* is a faulty component which is also capable of completing a task satisfactorily.
- *Res* is a resource: the faulty component may need to cooperate with a resource in order to complete its task.
- *Repman* represents a repairman: the component also needs to cooperate with a repairman in order to be rapaired.
- the *System* consists of two components competing for access to the resource and the repairman.

Example 9

- Consider the following PEPA model for *System*:

$$
\begin{aligned}
Comp &\stackrel{\text{def}}{=} (error, \epsilon).(repair, \rho).Comp + (task, \mu).Comp \\
Res &\stackrel{\text{def}}{=} (task, \top).(reset, r).Res \\
Repman &\stackrel{\text{def}}{=} (repair, \top).Repman \\
System &\stackrel{\text{def}}{=} ((Comp \| Comp) \underset{\{task\}}{\bowtie} Res) \underset{\{repair\}}{\bowtie} Repman
\end{aligned}
$$

Example 9

- Draw the derivation graph of *System*:

$$S_0 \stackrel{\text{def}}{=} ((Comp \| Comp) \underset{\{task\}}{\bowtie} Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_1 \stackrel{\text{def}}{=} ((Comp \| Comp) \underset{\{task\}}{\bowtie} (reset, r).Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_2 \stackrel{\text{def}}{=} (((repair, \rho).Comp \| Comp) \underset{\{task\}}{\bowtie} Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_3 \stackrel{\text{def}}{=} ((Comp \| (repair, \rho).Comp) \underset{\{task\}}{\bowtie} Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_4 \stackrel{\text{def}}{=} (((repair, \rho).Comp \| (repair, \rho).Comp) \underset{\{task\}}{\bowtie} Res) \underset{\{repair\}}{\bowtie} Repman$$

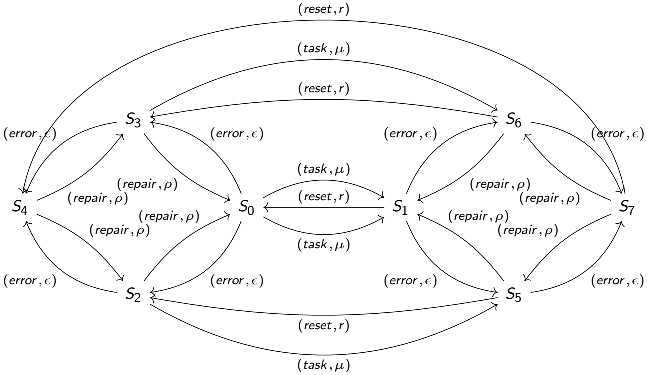
$$S_5 \stackrel{\text{def}}{=} (((repair, \rho).Comp \| Comp) \underset{\{task\}}{\bowtie} (reset, r).Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_6 \stackrel{\text{def}}{=} ((Comp \| (repair, \rho).Comp) \underset{\{task\}}{\bowtie} (reset, r).Res) \underset{\{repair\}}{\bowtie} Repman$$

$$S_7 \stackrel{\text{def}}{=} (((repair, \rho).Comp \| (repair, \rho).Comp) \underset{\{task\}}{\bowtie} (reset, r).Res) \underset{\{repair\}}{\bowtie} Repman$$

Example 9

- The derivation graph of *System* is:



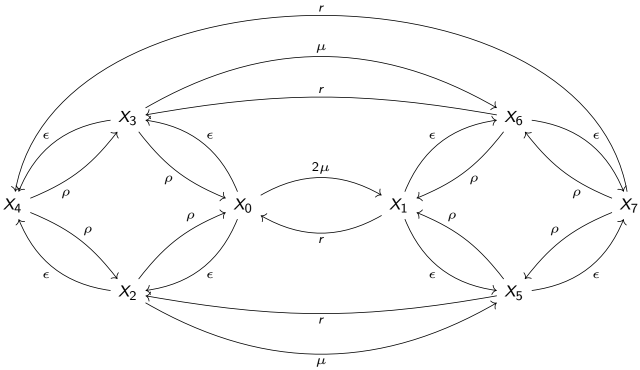
Strong equivalence and lumpability

Example 9

- Note that there is a pair of arcs in the derivation graph between the initial state S_0 and its one-step derivative S_1.
- This captures the fact that there are two distinct derivations of the activity $(task, \mu)$ according to whether the first or second component completes the task in cooperation with the resource.
- The derivation graph is the basis of the underlying CTMC.

Example 9

- The underlying CTMC is:



Example 9

- Consider the reflexive, symmetric and transitive closure of the following relation:

$$\mathcal{R} = \{(S_2, S_3), (S_5, S_6)\} \cup Id$$

whete Id is the identity relation.

- $\mathcal{R}$ induces the following equivalence classes:

$$[S_2] = \{S_2, S_3\} \quad [S_5] = \{S_5, S_6\} \quad [S_i] = \{S_i\} \text{ for } i \in \{0, 1, 4, 7\}$$

- We can prove that $\mathcal{R}$ is a strong equivalence.

Strong equivalence and lumpability

Example 9

- The aggregated CTMC is: