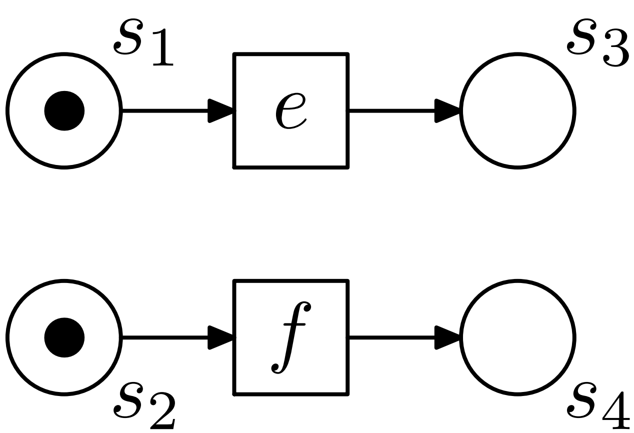
Synchronicity and concurrency in Petri Nets

Tomáš Šingliar, supervised by Gabriel Juhás

May 22, 2002

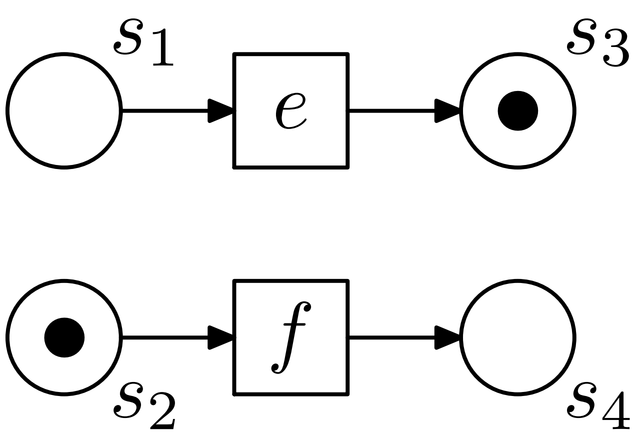
Place/Transition PN

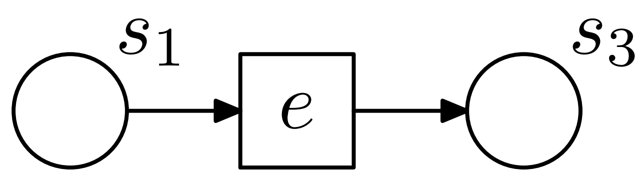
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Place/Transition

Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.

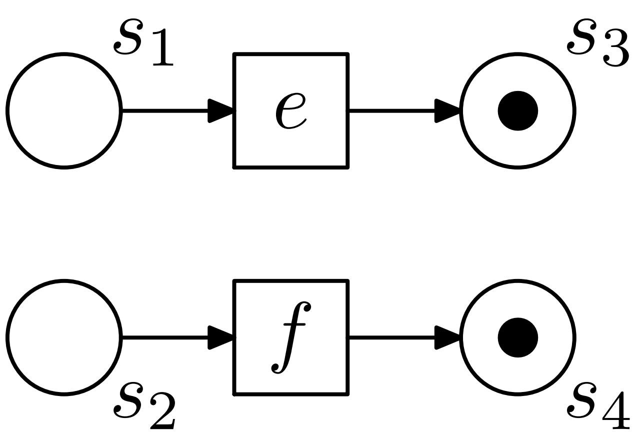




Firing sequences: $\{e\}$

Place/Transition

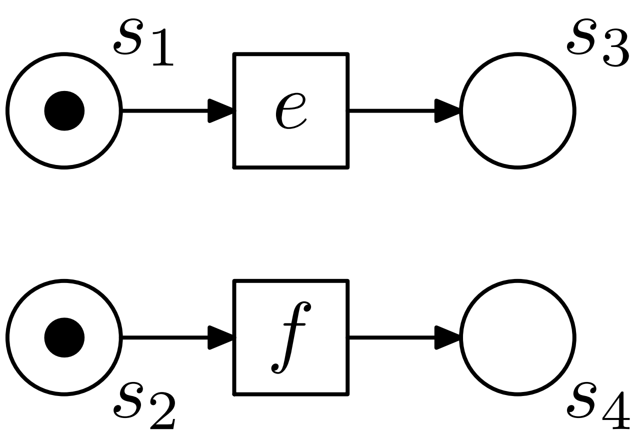
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$

Place/Transition

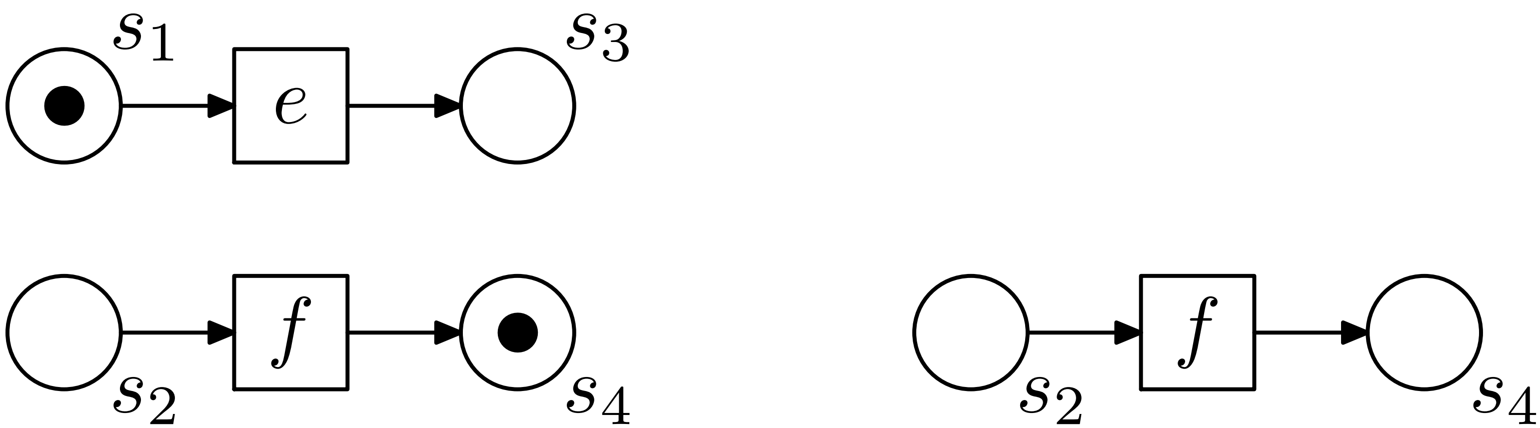
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$,

Place/Transition

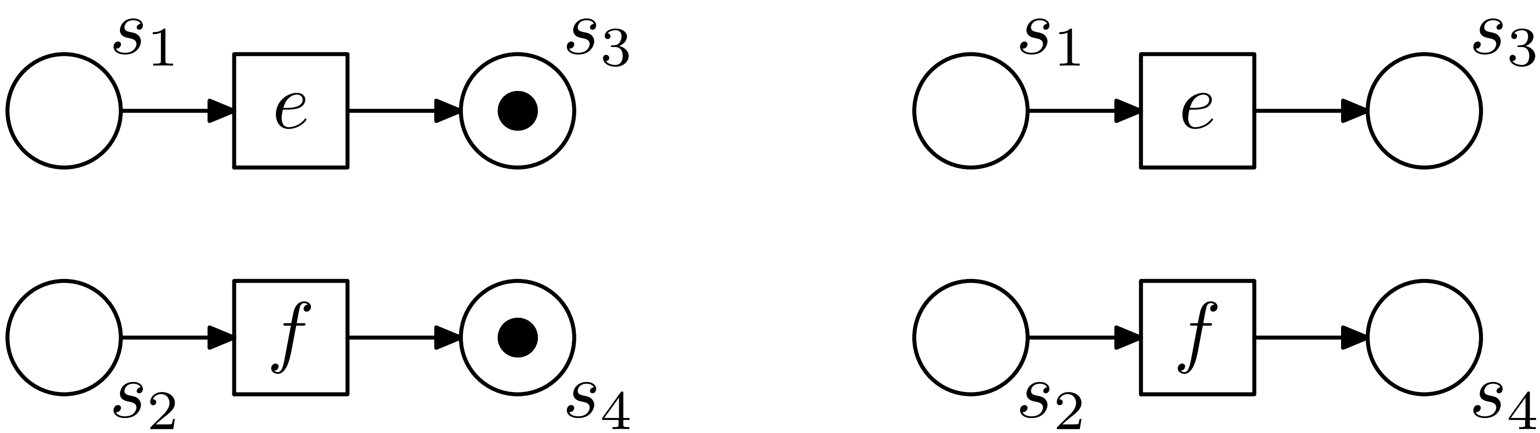
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$, $\{f\}$

Place/Transition

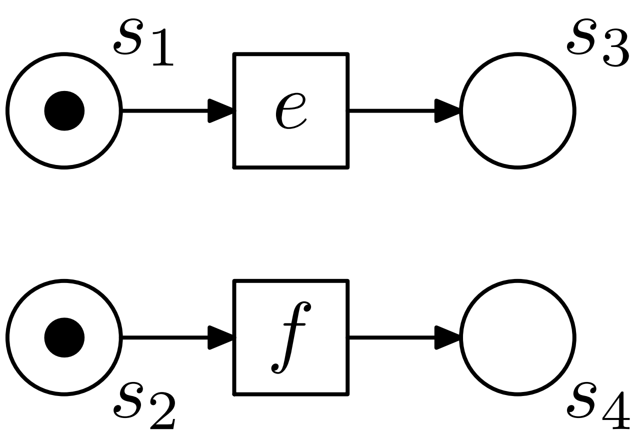
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$, $\{f\}\{e\}$

Place/Transition

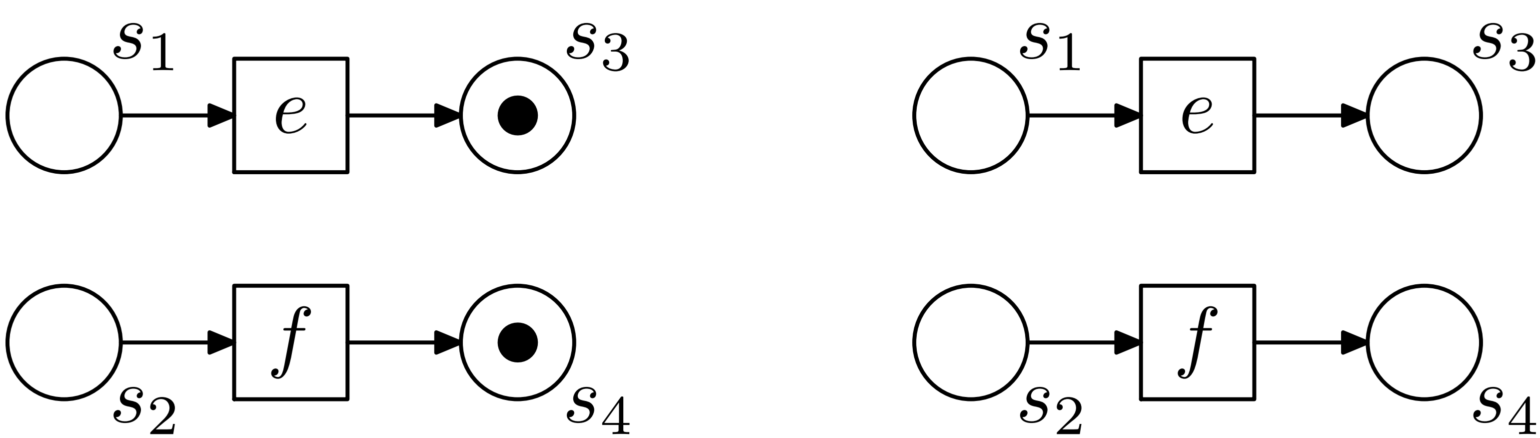
Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$, $\{f\}\{e\}$,

Place/Transition

Transitions not ordered by flow relation
$\Longrightarrow$ they can be fired in any order.



Firing sequences: $\{e\}\{f\}$, $\{f\}\{e\}$, $\{e, f\}$

Place/Transition Petri Nets

flow independence of transitions

$=$

they can fire in any order

Some extensions do not preserve the property

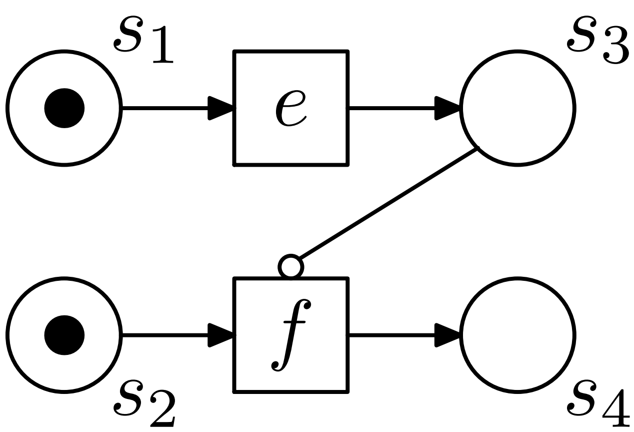
flow independence of transitions

$\neq$

they can fire in any order

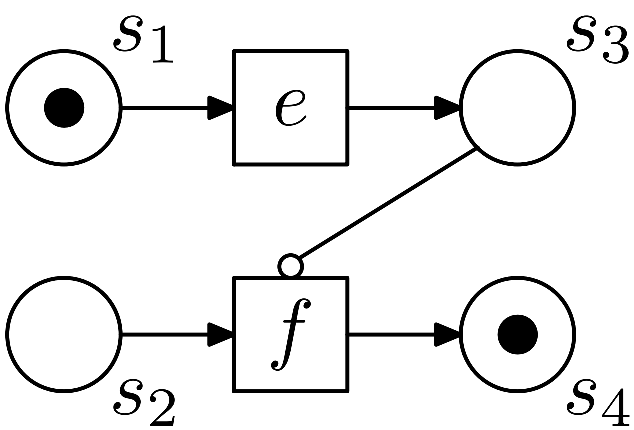
Petri Nets with Inhibitor Arcs

If a token is present in inhibitor place, transition is disabled.



Petri Nets with Inhibitor Arcs

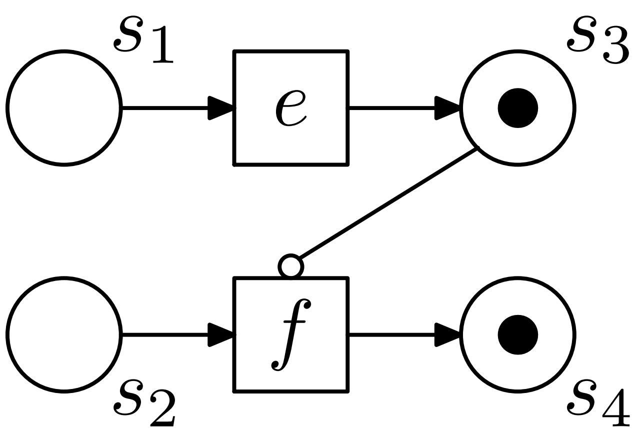
If a token is present in inhibitor place, transition is disabled.



Firing sequences: $\{f\}$

Petri Nets with Inhibitor Arcs

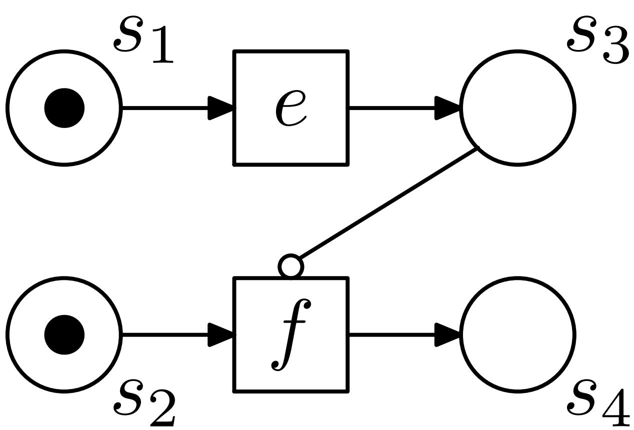
If a token is present in inhibitor place, transition is disabled.



Firing sequences: $\{f\}\{e\}$

Petri Nets with Inhibitor Arcs

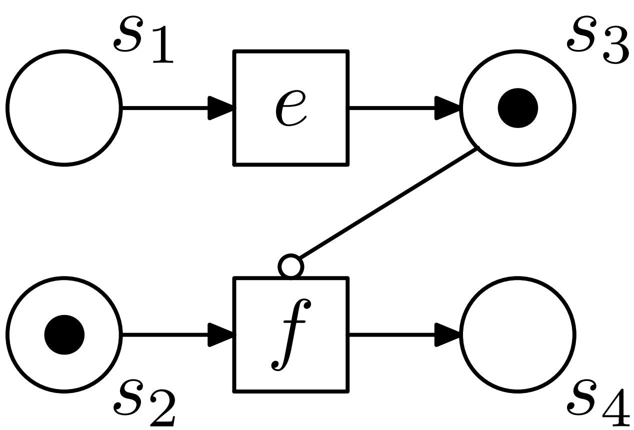
If a token is present in inhibitor place, transition is disabled.



Firing sequences: $\{f\}\{e\}$,

Petri Nets with Inhibitor Arcs

If a token is present in inhibitor place, transition is disabled.

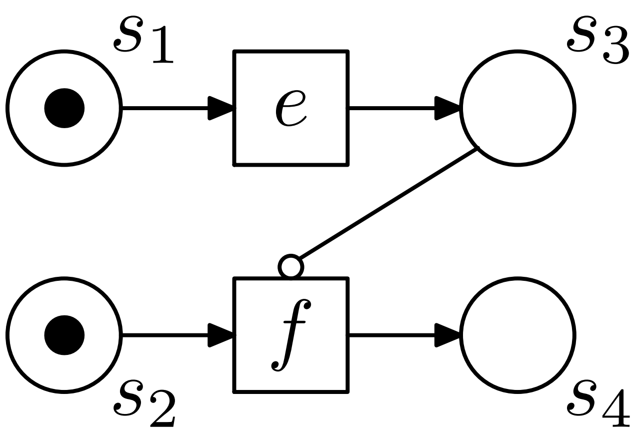


Firing sequences: $\{f\}\{e\}$, $\{e\}$

f is now disabled!

Petri Nets with Inhibitor Arcs

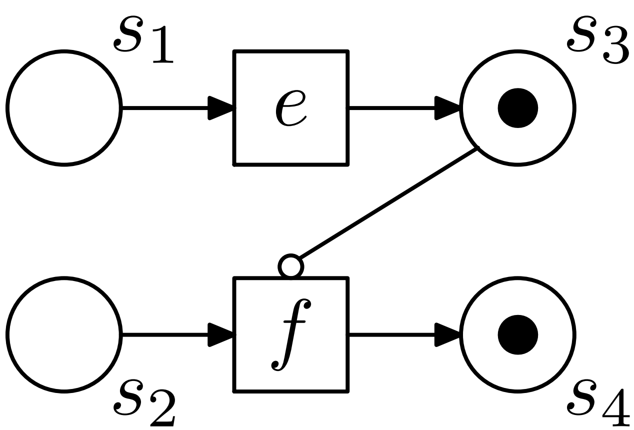
If a token is present in inhibitor place, transition is disabled.



Firing sequences: $\{f\}\{e\}$,

Petri Nets with Inhibitor Arcs

If a token is present in inhibitor place, transition is disabled.



Firing sequences: $\{f\}\{e\}$, $\{e, f\}$.

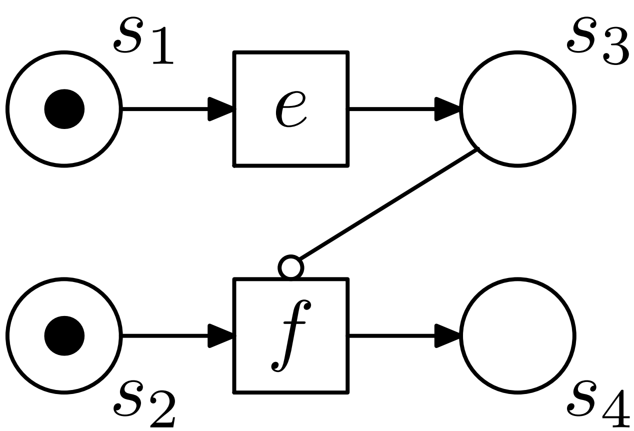
Inhibitors - Semantics

- If a token is present in inhibitor place, transition is disabled.

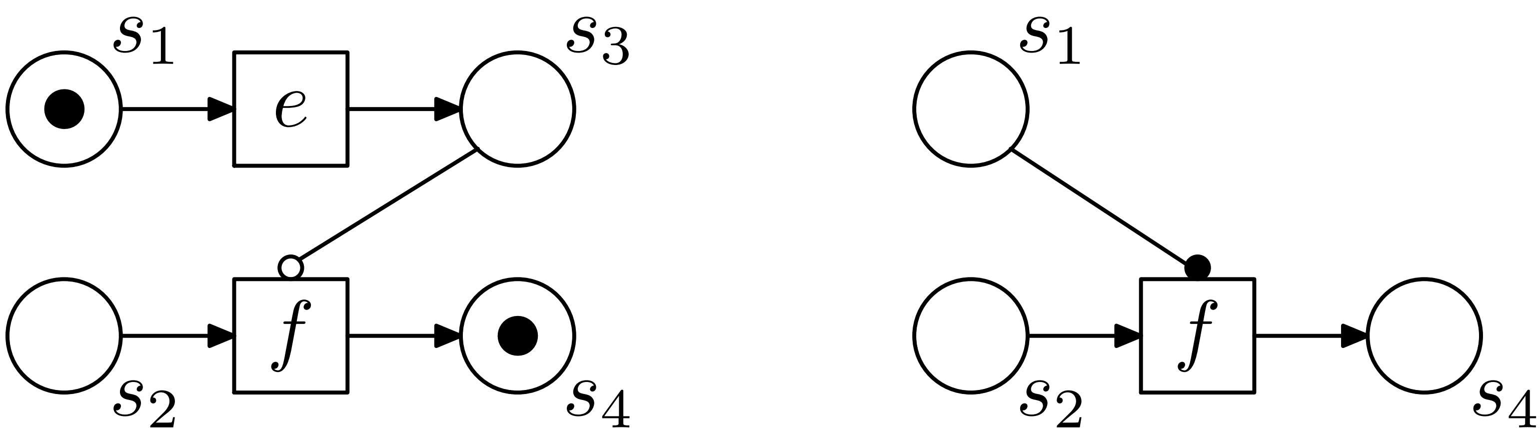
- In a process, place = *presence* of a token.
  Inhibitor arc tests for *absence* of a token.

- Test for presence of token in the *complement* place.

Inhibitor Arcs - Process Semantics

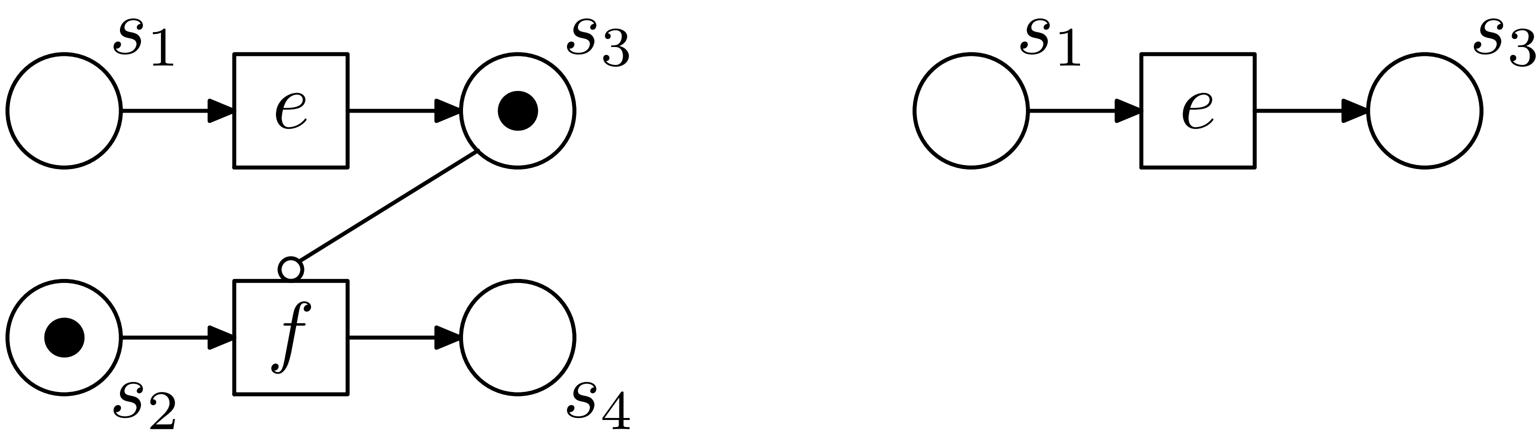


Inhibitor Arcs - Process Semantics



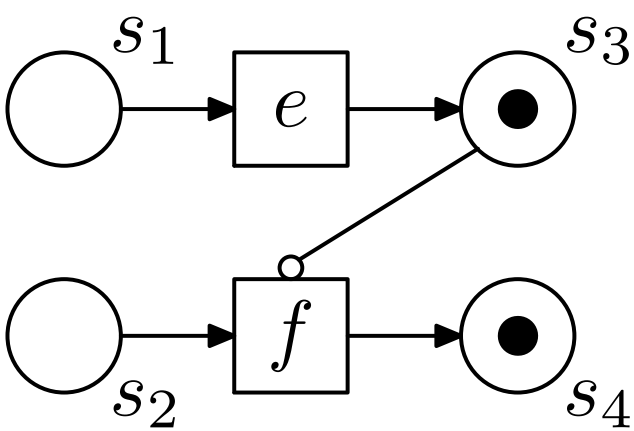
The place s_1 is complement of s_3

Inhibitor Arcs - Process Semantics



Nothing changes for the transition without an inhibitor arc

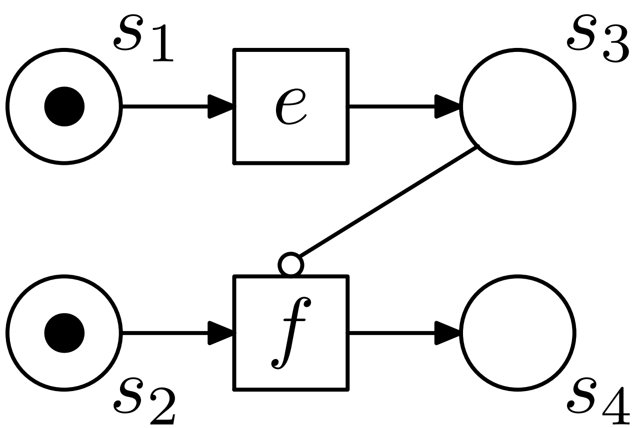
Inhibitor Arcs - Process Semantics



The resulting process - places labeled s_1 were glued

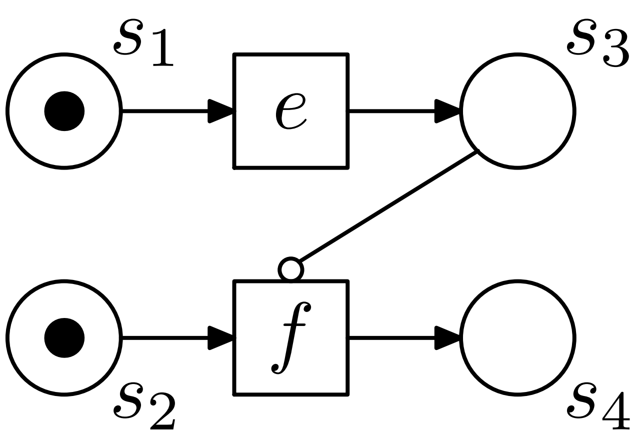
The corresponding partial order

Which partial order corresponds to this net ?



The corresponding partial order

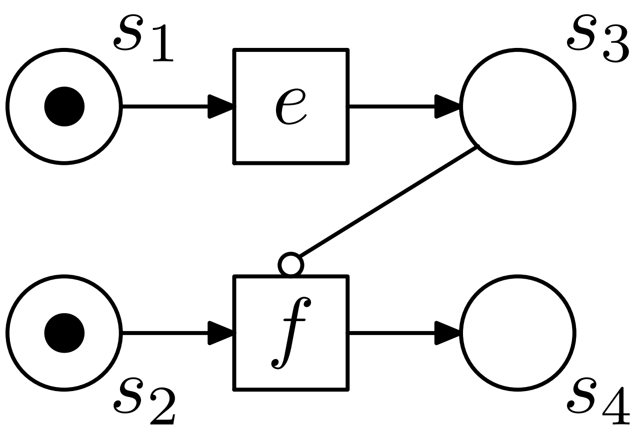
Which partial order corresponds to this net ?



$\emptyset$

The corresponding partial order

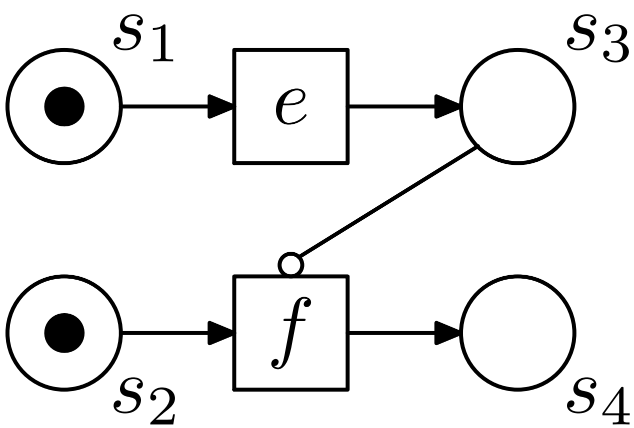
Which partial order corresponds to this net ?



$\emptyset$ $\{(f, e)\}$

The corresponding partial order

Which partial order corresponds to this net ?

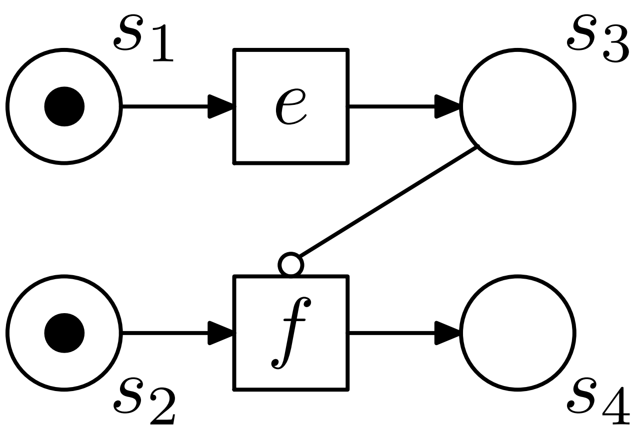


$\emptyset$ $\{(f,e)\}$

Inhibitor arcs under a-priori semantics bring new type of causality

The corresponding partial order

Which partial order corresponds to this net ?

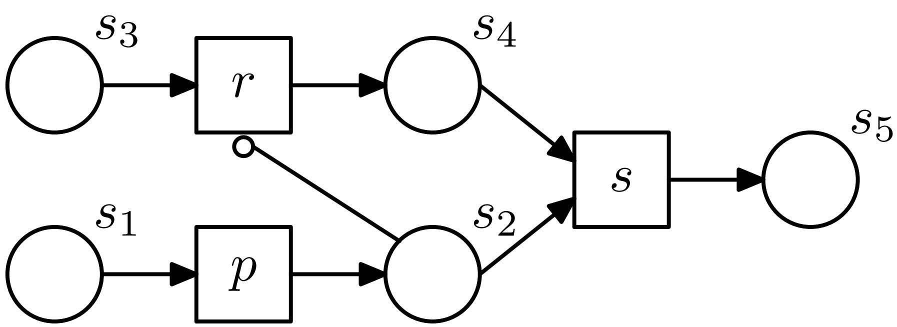


$\emptyset$ $\{(f, e)\}$

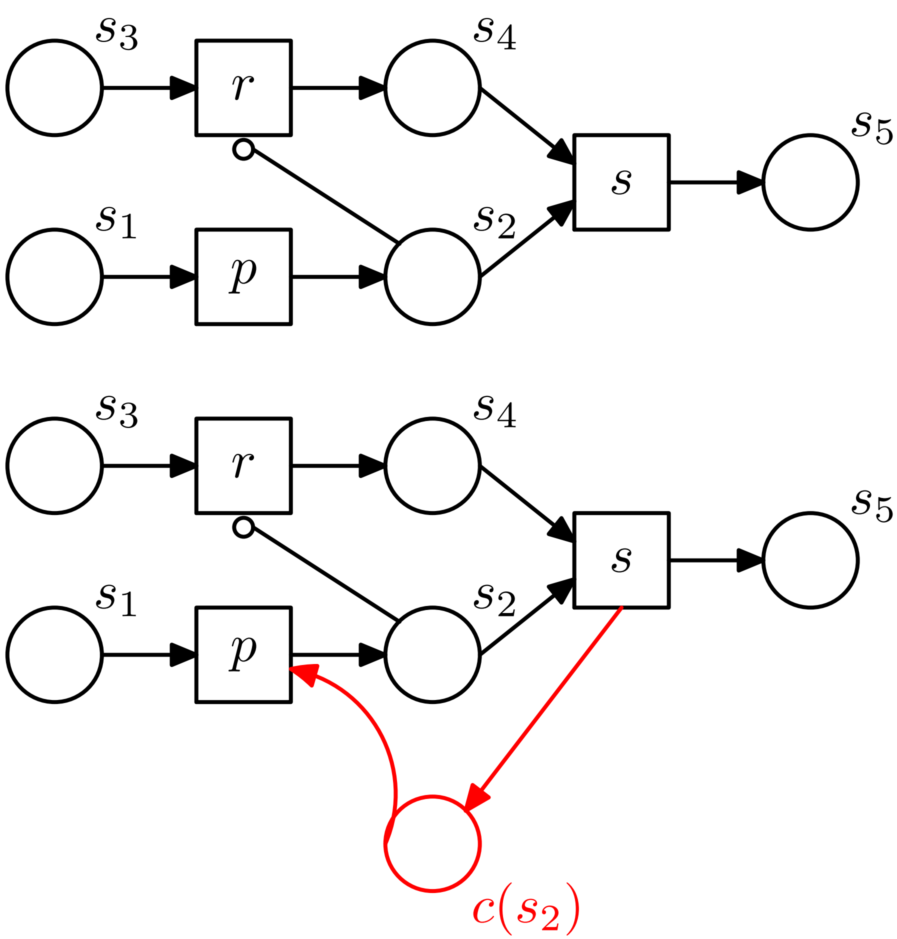
Inhibitor arcs under a-priori semantics bring new type of causality:

f must occur **not later than** e

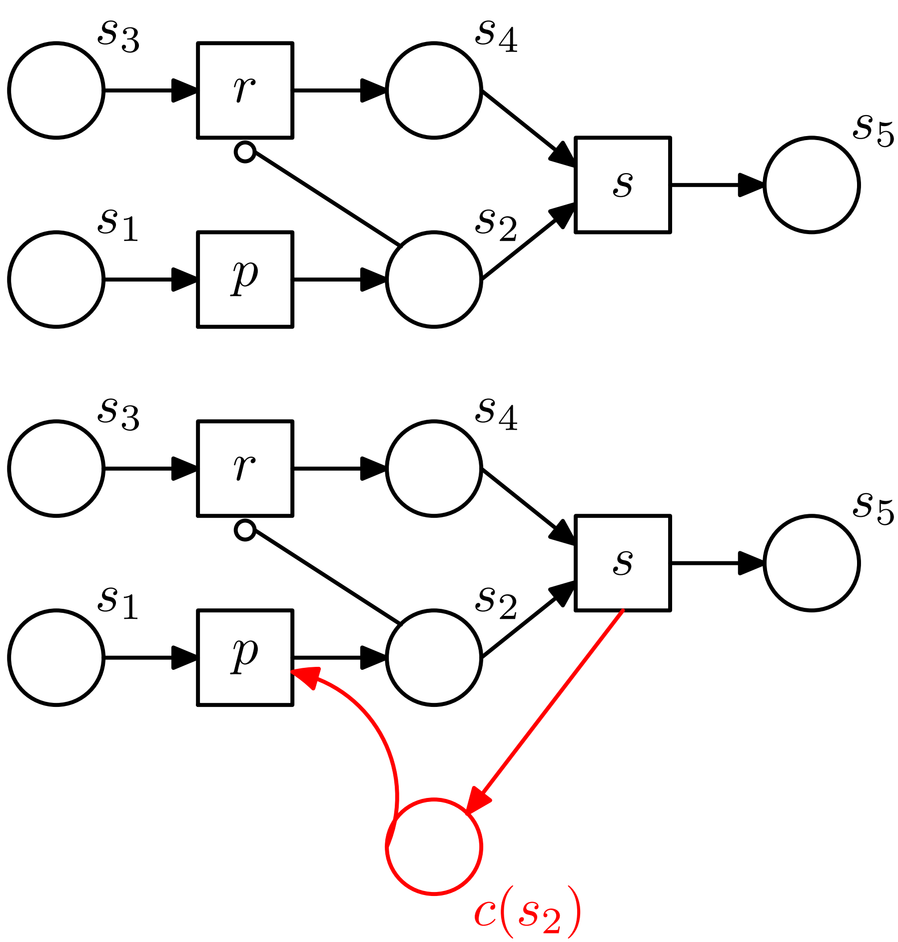
Stratified order structure

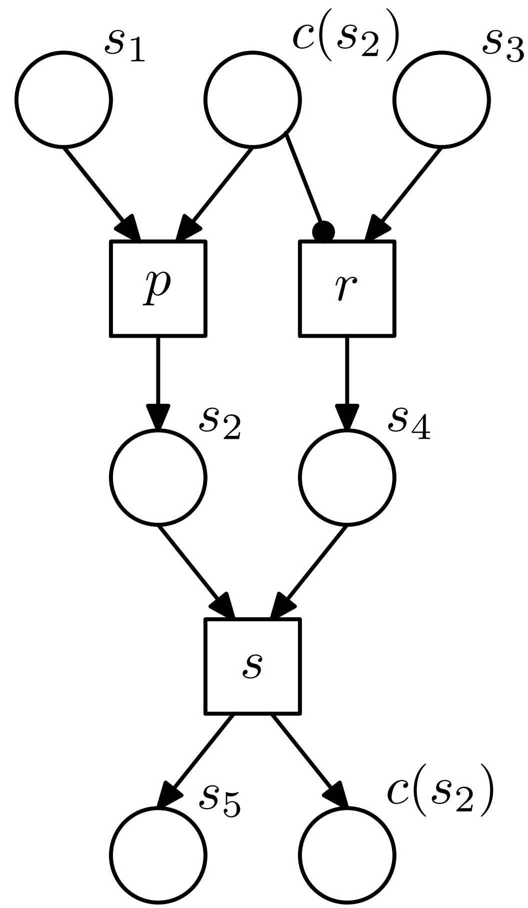


Stratified order structure

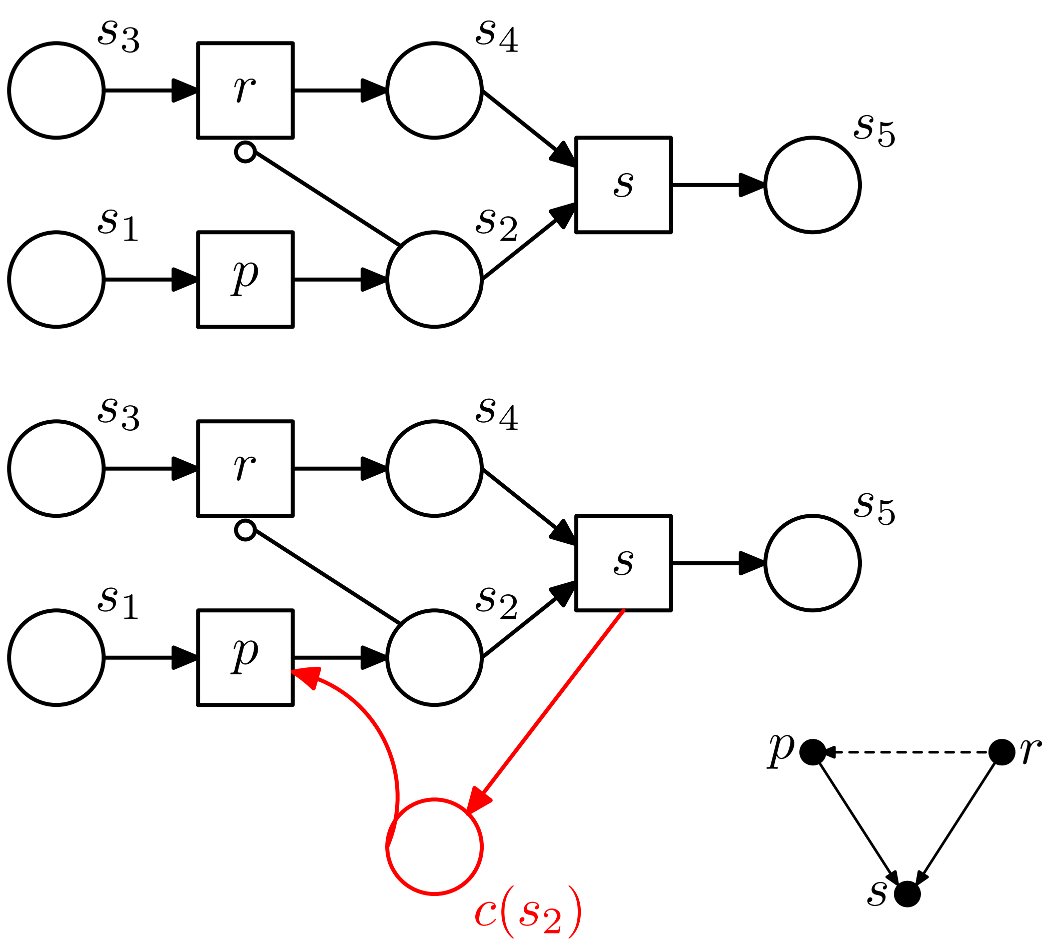


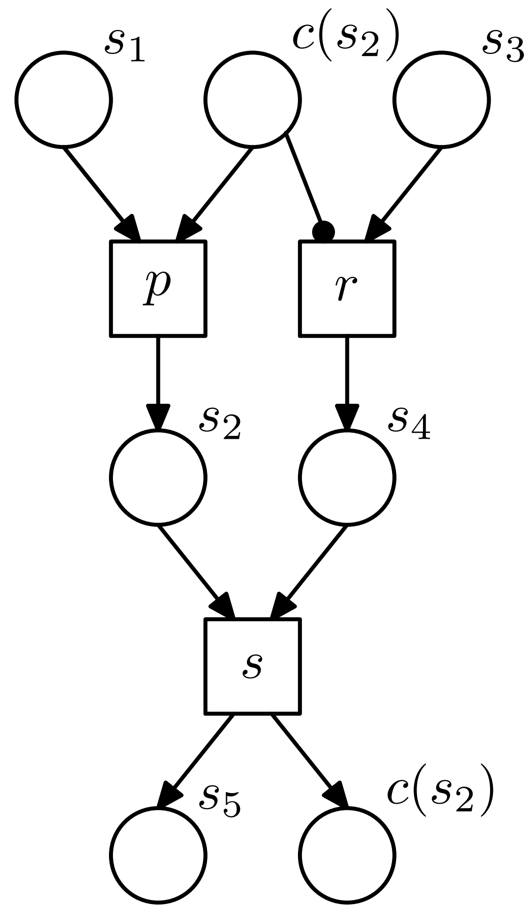
Stratified order structure





Stratified order structure





Stratified order structure

Janicky and Koutny discuss *stratified order structure*:

A *relational structure* is a triple $\mathcal{S} = (X, \prec, \sqsubset)$. $\mathcal{S}$ is called a *stratified order structure* if the following conditions are satisfied:

$$
\begin{array}{rcll}
 & & x \not\sqsubset x & (C1) \\
x \prec y & \Longrightarrow & x \sqsubset y & (C2) \\
x \sqsubset y \sqsubset z \;\wedge\; x \neq y & \Longrightarrow & x \sqsubset z & (C3) \\
x \sqsubset y \prec z \;\vee\; x \prec y \sqsubset z & \Longrightarrow & x \prec z & (C4)
\end{array}
$$

Resembles $<$ and $\leq$ on natural numbers (!)

Expressing behaviour by process terms

The set of process terms $\mathcal{P}$:

- Every elementary marking $m \in \mathcal{P}$
- Every transition $t \in \mathcal{P}$
- $\alpha, \beta \in \mathcal{P} \wedge \alpha, \beta$ independent $\implies \alpha \,\|\, \beta \in \mathcal{P}$
- $\alpha, \beta \in \mathcal{P} \wedge \alpha, \beta$ compatible $\implies \alpha ; \beta \in \mathcal{P}$

Expressing behaviour by process terms - Information

- Independence depends on places that the process (term) uses
- Assign each term information
- Usually Relation over sets of places

Expressing behaviour by process terms - Information

- Compatibility depends on places that the process (term) uses

- Assign each term information (structured)

- Relation over sets of places (tuples of sets)

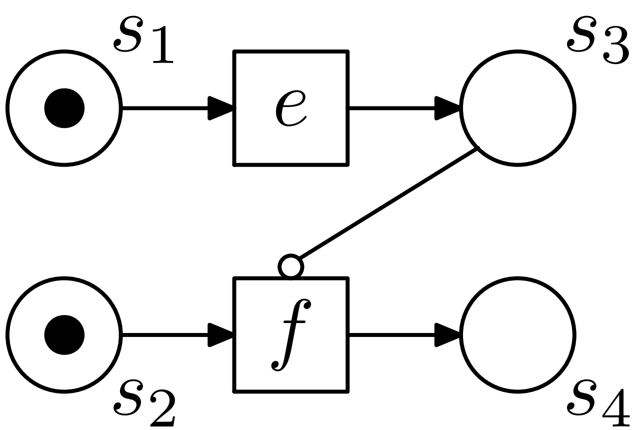
- Operations over information elements correspond to $\|$, ;

Expressing behaviour by process terms

Two step construction

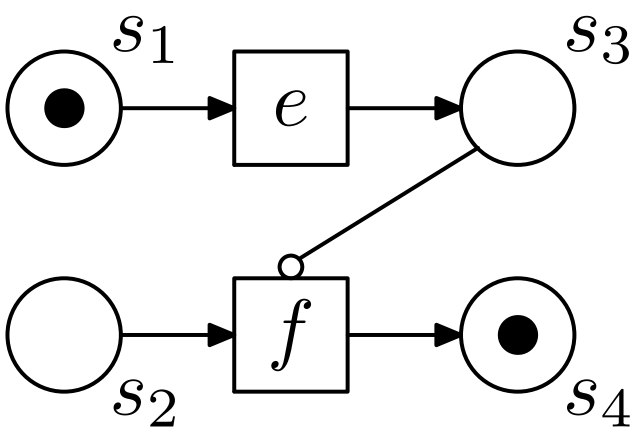
- Compose transitions into synchronous steps with $\oplus$
- Elementary terms are now
  - markings
  - synchronous steps

Process terms - Example



Process terms: $id_{\{s_1,s_2\}}$

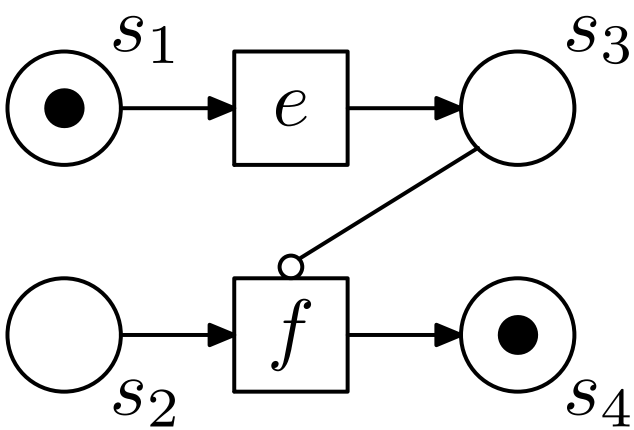
Process terms - Example



Firing sequences: $\{f\}$

Process terms: $id_{\{s_1,s_2\}}$, f

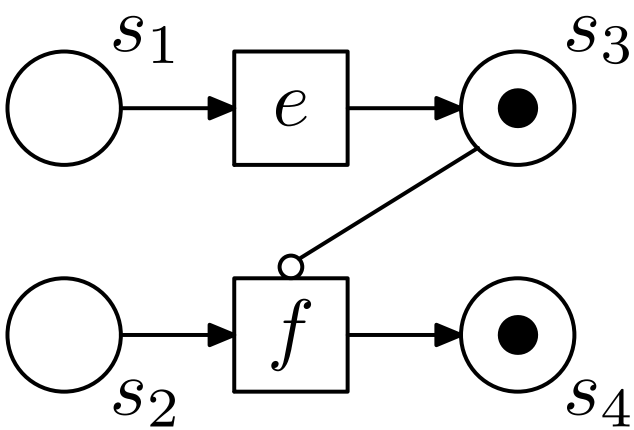
Process terms - Example



Firing sequences: $\{f\}$

Process terms: $id_{\{s_1,s_2\}}$, f;

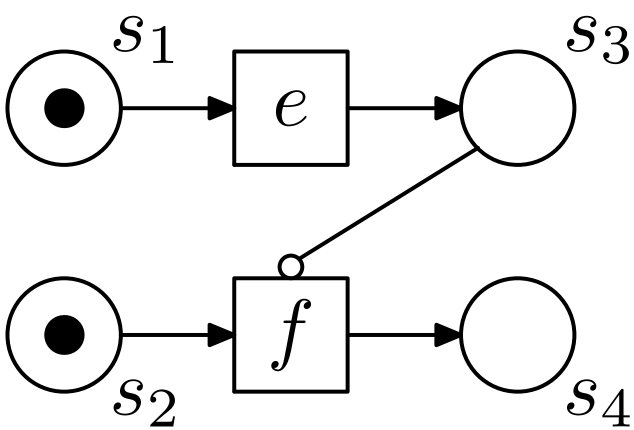
Process terms - Example



Firing sequences: $\{f\}\{e\}$

Process terms: $id_{\{s_1,s_2\}}$, $f;e$

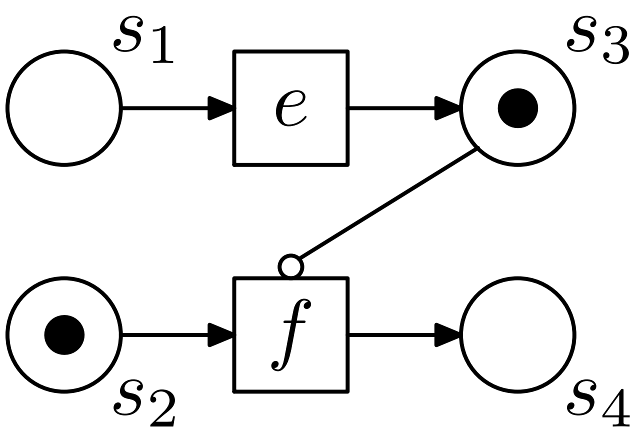
Process terms - Example



Firing sequences: $\{f\}\{e\}$,

Process terms: $id_{\{s_1,s_2\}}$, $f;e$,

Process terms - Example

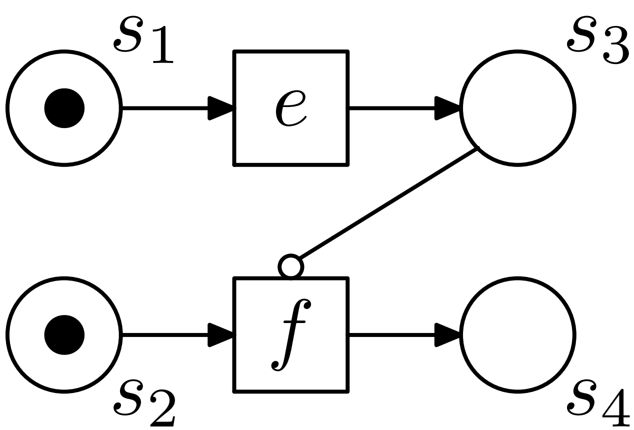


Firing sequences: $\{f\}\{e\}$, $\{e\}$

Process terms: $id_{\{s_1,s_2\}}$, $f;e$, $e;f$ is undefined

f is now disabled!

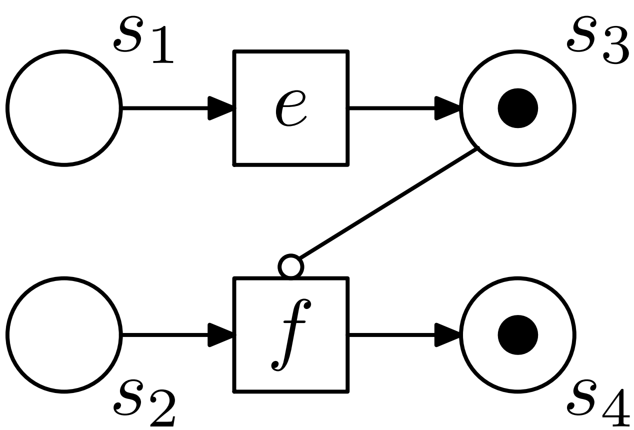
Process terms - Example



Firing sequences: $\{f\}\{e\}$,

Process terms: $id_{\{s_1,s_2\}}$, $f;e$,

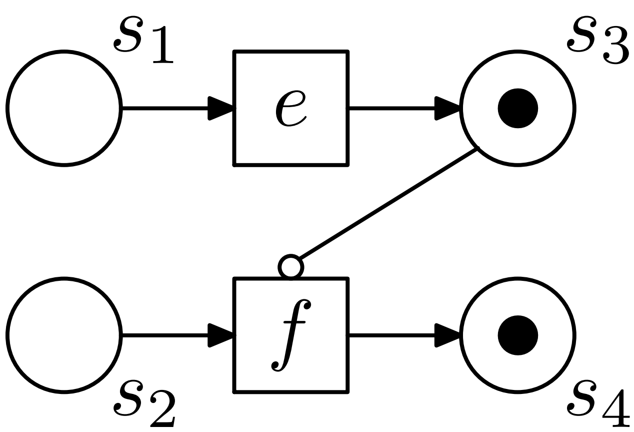
Process terms - Example



Firing sequences: $\{f\}\{e\}$, $\{e, f\}$

Process terms: $id_{\{s_1,s_2\}}$, $f;e$, $e \oplus f$

Process terms - Example



Firing sequences: $\{f\}\{e\}$, $\{e, f\}$

Process terms: $id_{\{s_1,s_2\}}$, $f;e$, $e \oplus f$, $id_{\{s_3,s_4\}}$

Process equivalence $\sim$

Some terms express the same process

1. $\alpha \parallel \beta \sim \beta \parallel \alpha$
2. $(\alpha \parallel \beta) \parallel \gamma \sim \alpha \parallel (\beta \parallel \gamma)$
3. $(\alpha; \beta); \gamma \sim \alpha; (\beta; \gamma)$
4. $((\alpha_1 \parallel \alpha_2); (\alpha_3 \parallel \alpha_4)) \sim ((\alpha_1; \alpha_3) \parallel (\alpha_2; \alpha_4))$
5. $\alpha \oplus \beta \sim \beta \oplus \alpha$
6. $(\alpha \oplus \beta) \oplus \gamma \sim \alpha \oplus (\beta \oplus \gamma)$
7. $(\alpha \oplus \beta) \sim (\alpha \parallel pre(\beta)); (post(\alpha) \parallel \beta)$
8. $(\alpha; post(\alpha)) \sim \alpha \sim (pre(\alpha); \alpha)$
9. $id_{(m+n)} \sim id_m \parallel id_n$
10. $(\alpha + id_\emptyset) \sim \alpha$

Process equivalence $\sim$

Some terms express the same process

1. $\alpha \parallel \beta \sim \beta \parallel \alpha$
2. $(\alpha \parallel \beta) \parallel \gamma \sim \alpha \parallel (\beta \parallel \gamma)$
3. $(\alpha; \beta); \gamma \sim \alpha; (\beta; \gamma)$
4. $((\alpha_1 \parallel \alpha_2); (\alpha_3 \parallel \alpha_4)) \sim ((\alpha_1; \alpha_3) \parallel (\alpha_2; \alpha_4))$
5. $\alpha \oplus \beta \sim \beta \oplus \alpha$
6. $(\alpha \oplus \beta) \oplus \gamma \sim \alpha \oplus (\beta \oplus \gamma)$
7. $(\alpha \oplus \beta) \sim (\alpha \parallel pre(\beta)); (post(\alpha) \parallel \beta)$
8. $(\alpha; post(\alpha)) \sim \alpha \sim (pre(\alpha); \alpha)$
9. $id_{(m+n)} \sim id_m \parallel id_n$
10. $(\alpha + id_\emptyset) \sim \alpha$

Axiom 4 is key in proof by Desel, Juhas and Lorenz

Process equivalence $\sim$

Some terms express the same process

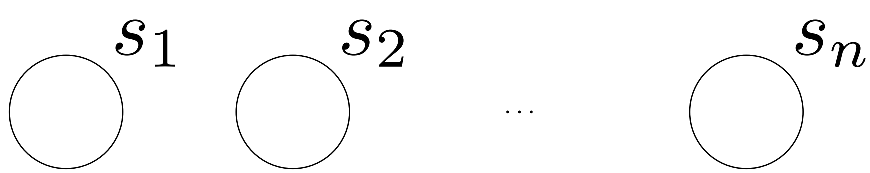
1. $\alpha \,\|\, \beta \sim \beta \,\|\, \alpha$
2. $(\alpha \,\|\, \beta) \,\|\, \gamma \sim \alpha \,\|(\beta \,\|\, \gamma)$
3. $(\alpha; \beta); \gamma \sim \alpha; (\beta; \gamma)$
4. $((\alpha_1 \,\|\, \alpha_2); (\alpha_3 \,\|\, \alpha_4)) \sim ((\alpha_1; \alpha_3) \,\|(\alpha_2; \alpha_4))$
5. $\alpha \oplus \beta \sim \beta \oplus \alpha$
6. $(\alpha \oplus \beta) \oplus \gamma \sim \alpha \oplus (\beta \oplus \gamma)$
7. $(\alpha \oplus \beta) \sim (\alpha \,\|\, pre(\beta)); (post(\alpha) \,\|\, \beta)$
8. $(\alpha; post(\alpha)) \sim \alpha \sim (pre(\alpha); \alpha)$
9. $id_{(m+n)} \sim id_m \,\|\, id_n$
10. $(\alpha + id_{\emptyset}) \sim \alpha$

Axiom 7 relates "classical" operations $\|$ and ; to $\oplus$

The corresponding process

Defined similarly to the case of EN

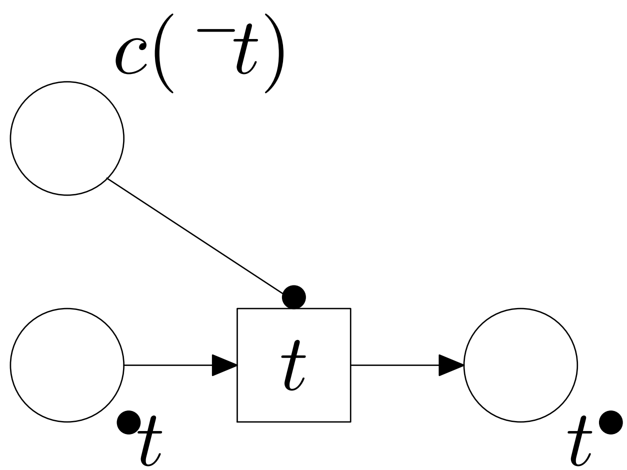
- Elementary marking $m = \{s_1, \ldots, s_n\}$



The corresponding process

Defined similarly to the case of EN

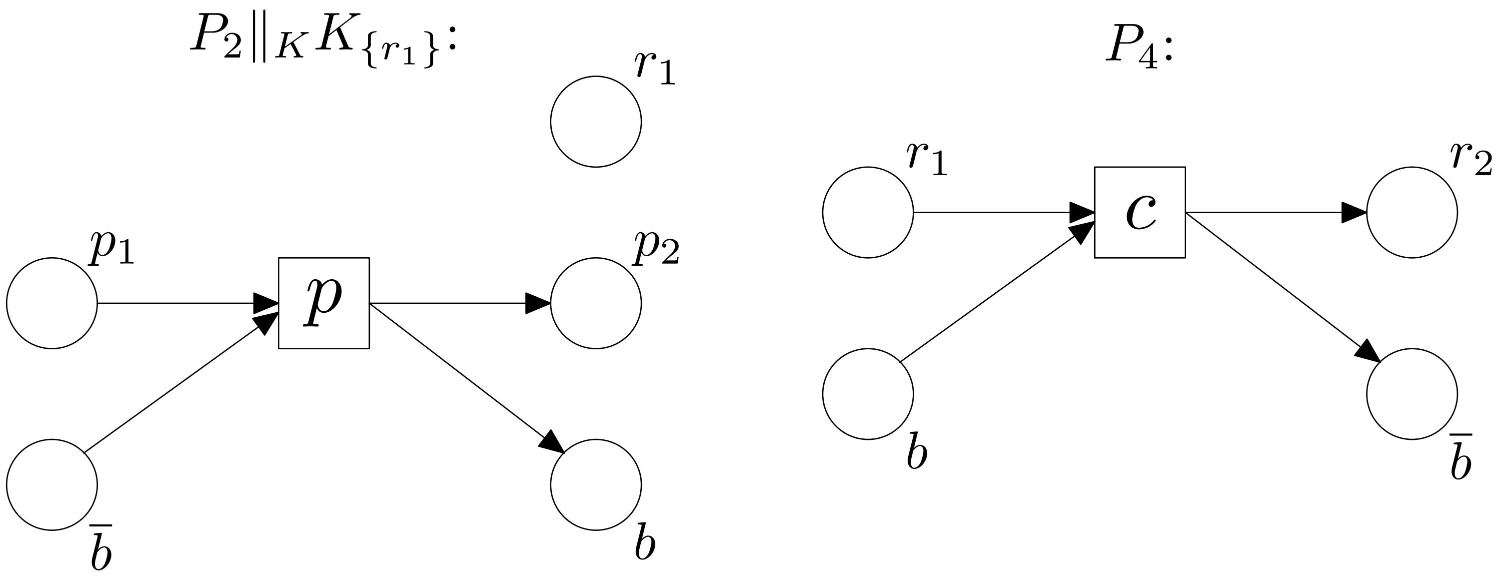
- Transition t



The corresponding process

Defined similarly to the case of EN

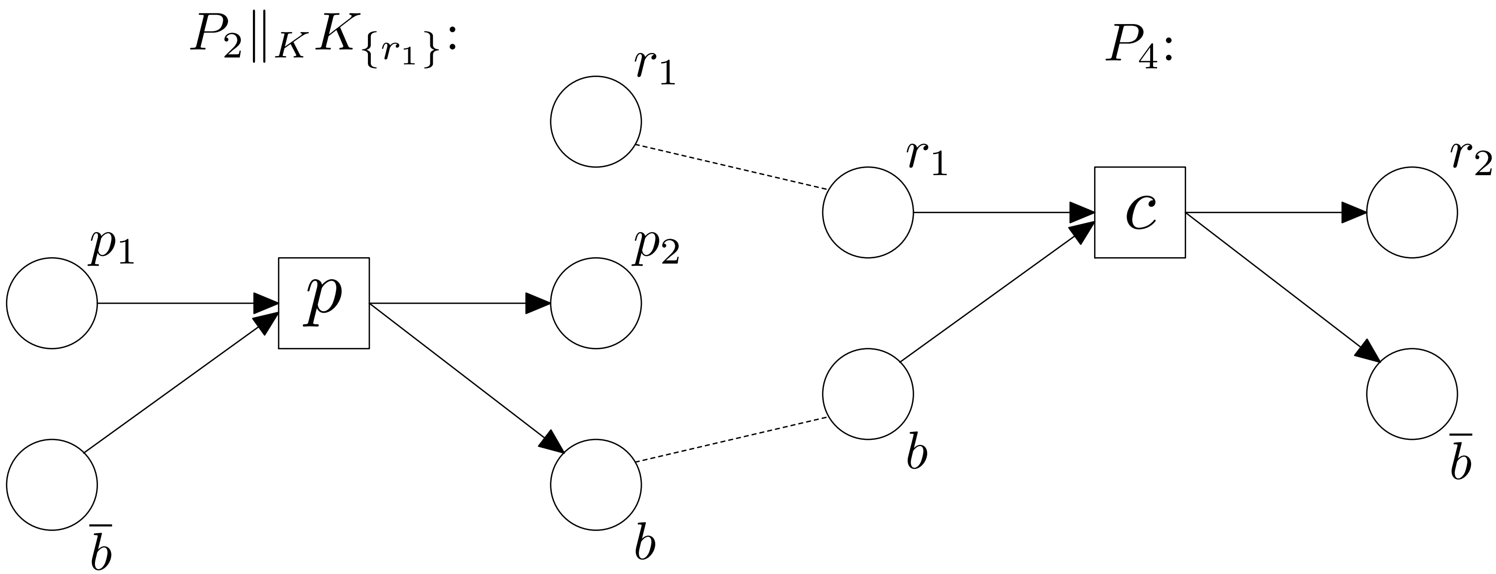
- Sequential composition



The corresponding process

Defined similarly to the case of EN

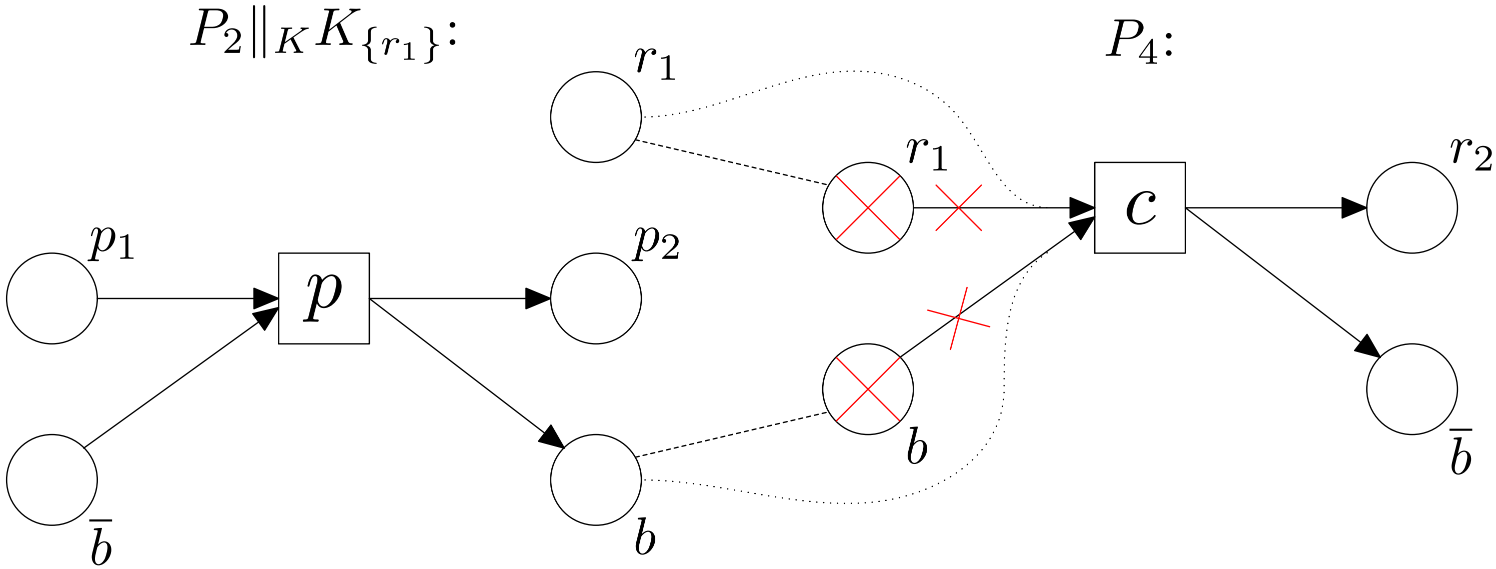
- Sequential composition



The corresponding process

Defined similarly to the case of EN

- Sequential composition

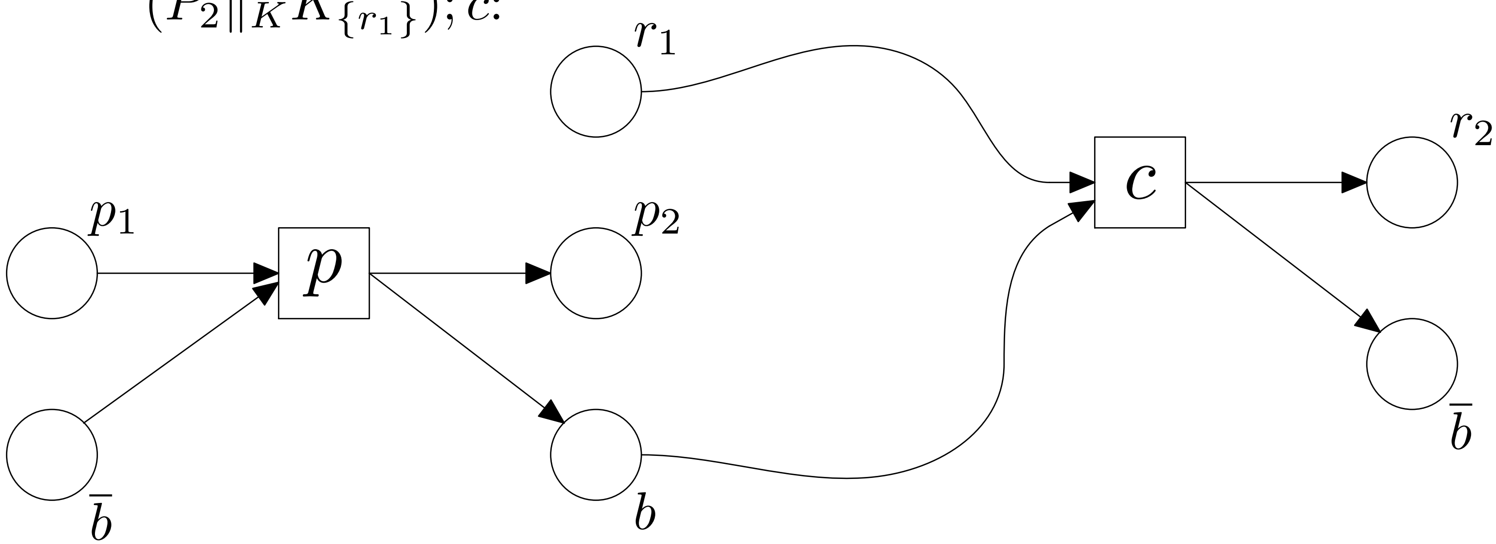


The corresponding process

Defined similarly to the case of EN

- Sequential composition

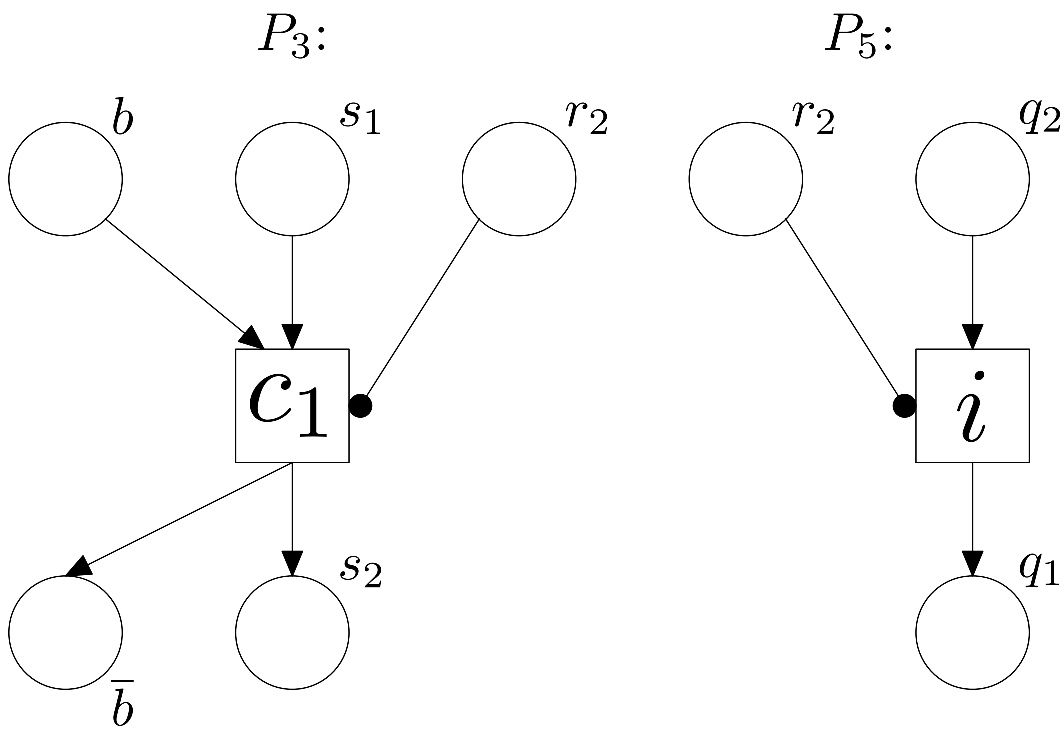
$(P_2 \|_K K_{\{r_1\}}); c$:



The corresponding process

Defined similarly to the case of EN

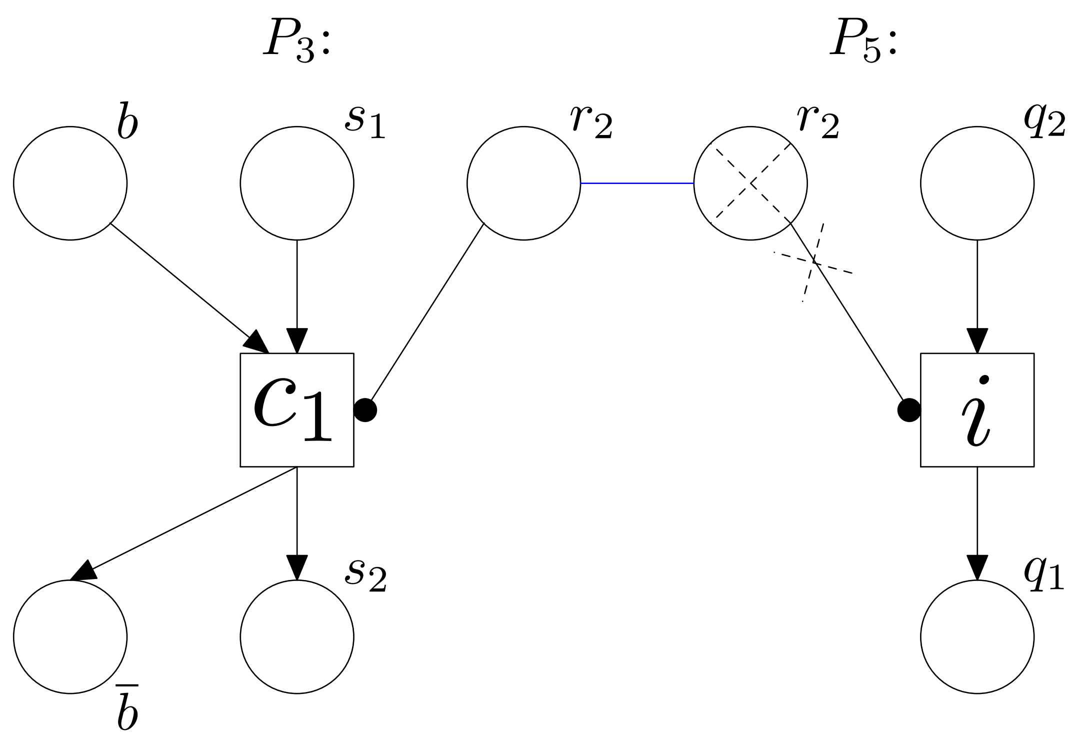
- Concurrent composition



The corresponding process

Defined similarly to the case of EN

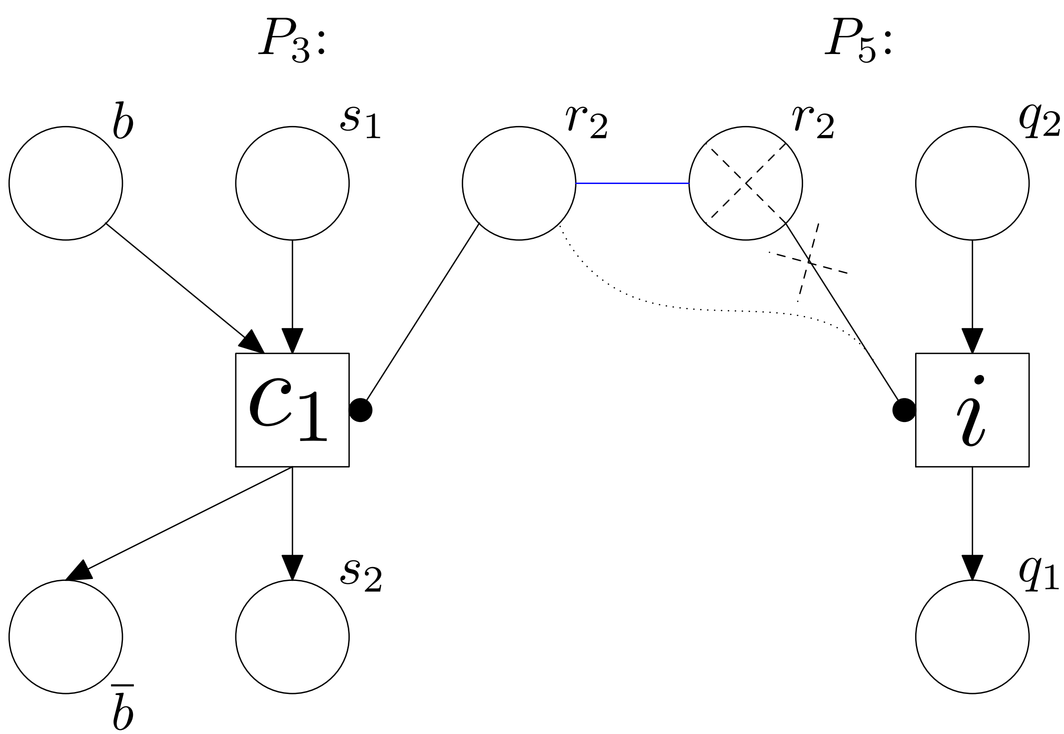
- Concurrent composition



The corresponding process

Defined similarly to the case of EN

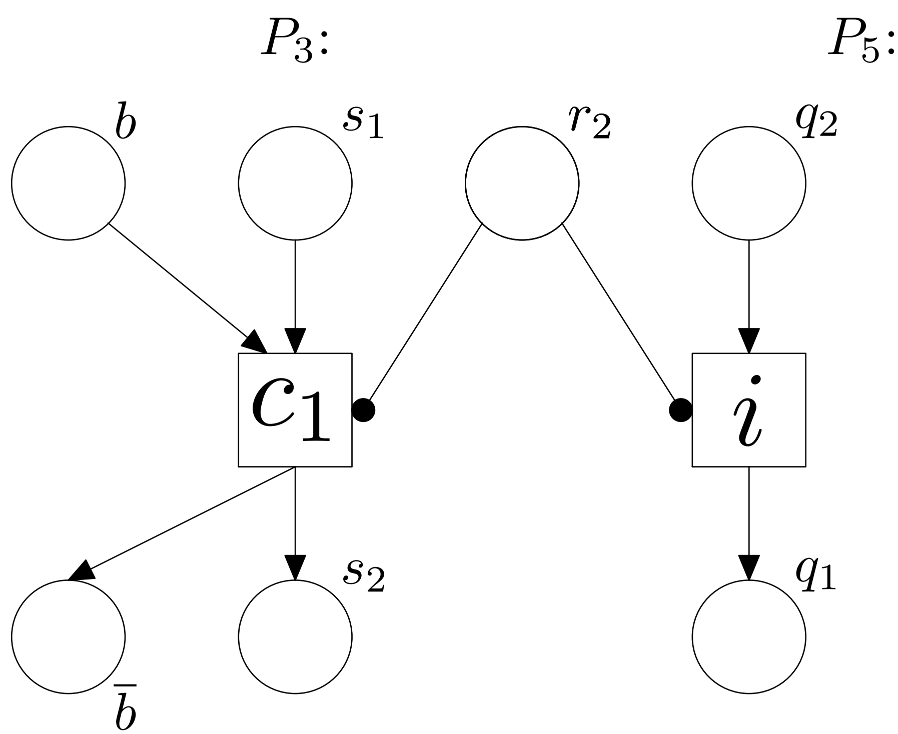
- Concurrent composition



The corresponding process

Defined similarly to the case of EN

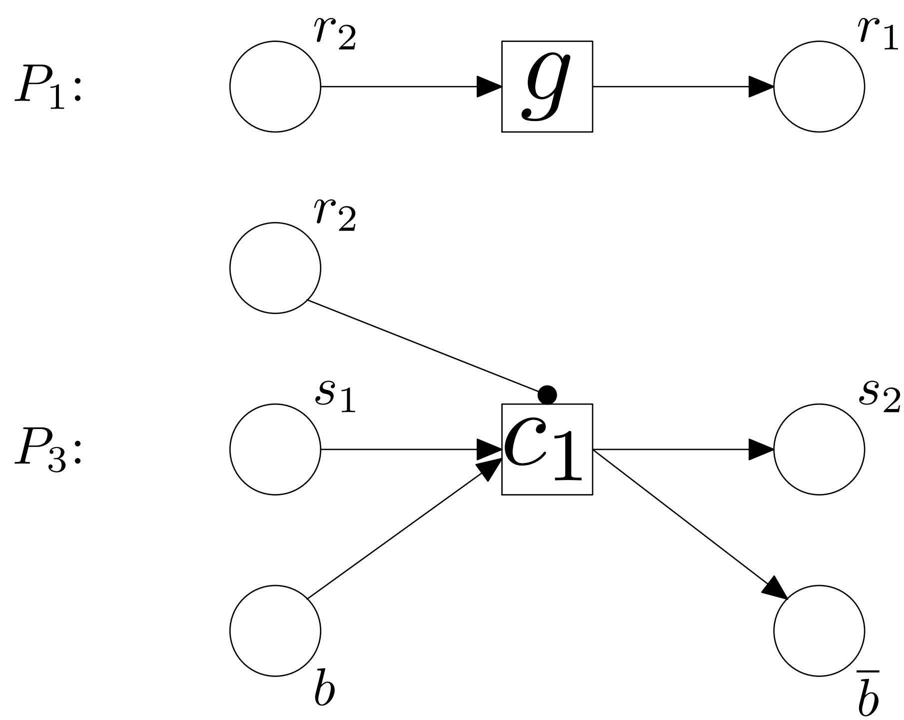
- Concurrent composition



The corresponding process

Defined similarly to the case of EN

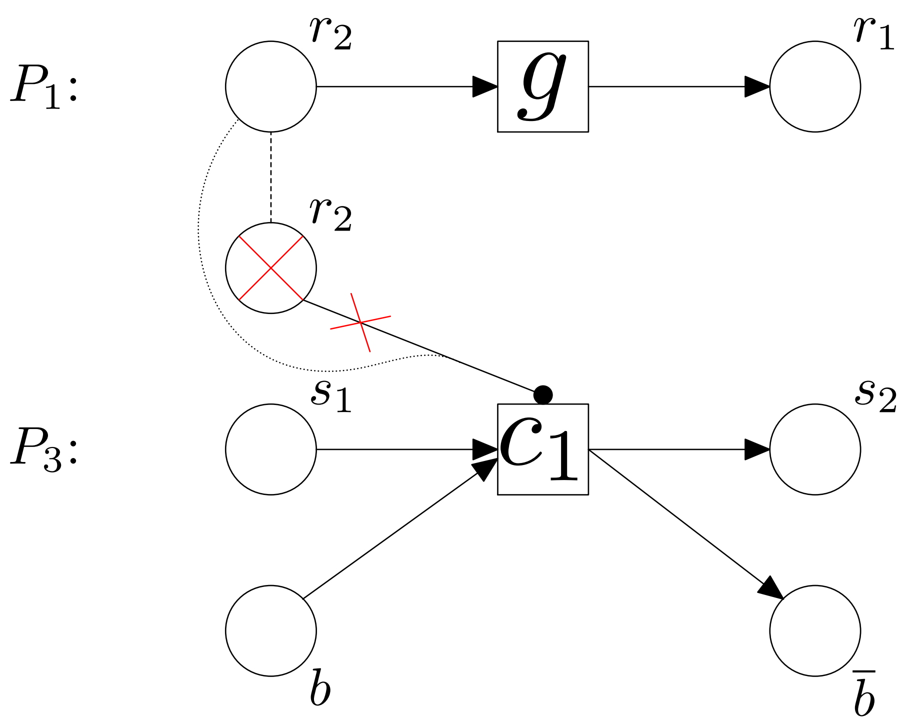
- Synchronous composition



The corresponding process

Defined similarly to the case of EN

- Synchronous composition

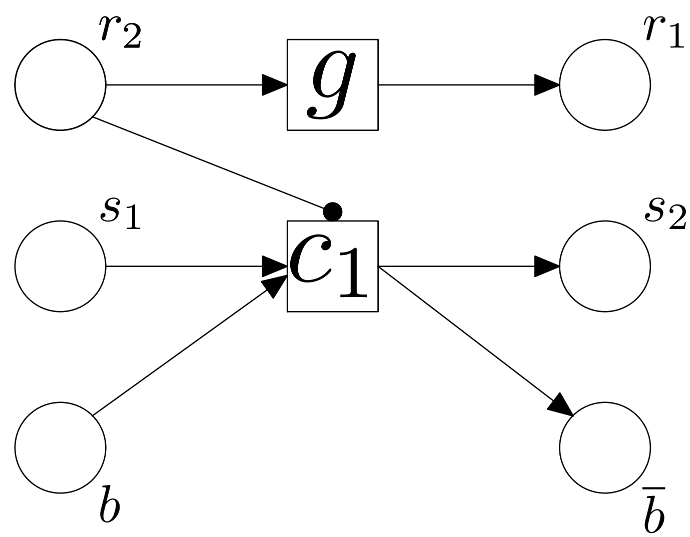


The corresponding process

Defined similarly to the case of EN

- Synchronous composition

$P_3 \oplus P_1$:



The corresponding process

We formalize the operation of process formation and denote it τ

$$\tau : \mathcal{P} \longrightarrow proc_{ao}$$

The results - Overview

It holds

- τ is surjective, i.e. every process corresponds to a term
- $\alpha \sim \beta \implies \tau(\alpha)$ is isomorphic to $\tau(\beta)$
- $\tau(\alpha)$ is isomorphic to $\tau(\beta) \implies \alpha \sim \beta$
- Corollary: There is one-to-one correspondence between processes and $\sim$-equivalence classes of terms

The results - Surjectivity

Theorem: τ is surjective, every process corresponds to a term.

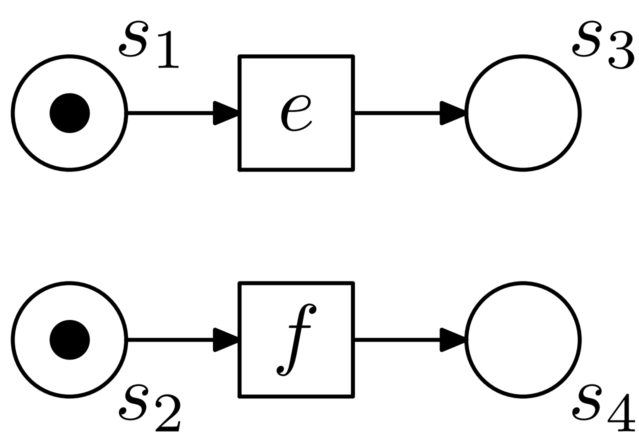
- Fire all enabled transitions synchronously, obtain S_1
- Iterate until final marking is reached
- $\left(\bigoplus_{t \in S_1} t\right); \ldots; \left(\bigoplus_{t \in S_k} t\right)$ is the term sought
- the term is in form of *maximal synchronous steps*

The results - Easy direction of equivalence

Theorem: $\alpha \sim \beta \implies \tau(\alpha)$ is isomorphic to $\tau(\beta)$

- Mathematically very technical and boring: For each axiom (10) and term construction rule (5) verify the claim

- Visually obvious: $e \,\|\, f$ and $f \,\|\, e$ give the same "picture"



On ordering

- The ordering of transitions in a process is only causal
- Valid terms honor causal dependencies
- But may introduce some “time sequences” that are not causal
- The proof more or less makes these hidden sequences explicit

The results - The difficult direction

Theorem: $\tau(\alpha)$ is isomorphic to $\tau(\beta) \Longrightarrow \alpha \sim \beta$.
Proof. Adopted the idea from DJL:

- The two terms have the same process K and contain the same set of transitions.

- Therefore they have the same maximal-step term γ, corresponding to K. Let $\gamma = \gamma^1; \ldots; \gamma^k$, γ is of form $t_1 \parallel t_2 \parallel \ldots t_k$.

- For every term, there is a sequentialization, e.g. $e \parallel f \rightarrow e; f$ or $f; e$

The results - The difficult direction

- Valid terms honor causal dependencies:
  $t_1; \ldots ; t_k$ is a valid term and $t_k \prec t_1$ cannot happen

- t_1 and t_k can be exchanged, within $\sim$-equivalence class

- By successive exchanges, we form γ^1 (axiom 4)

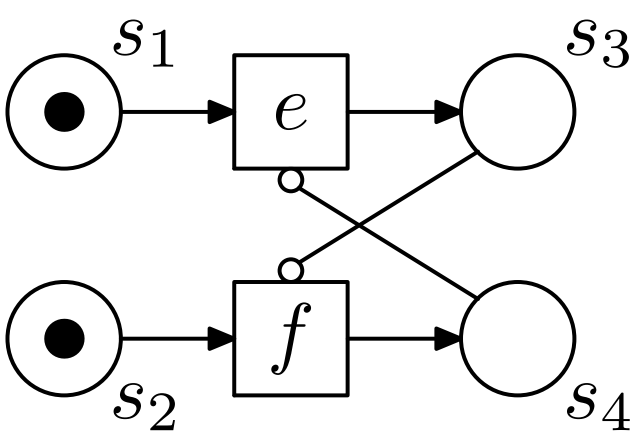
- Apply the process to the rest of term, obtain $\alpha \sim \gamma$

- By symmetry, $\beta \sim \gamma$. By transitivity $\alpha \sim \beta$.

It doesn't work.

It doesn't work.

For Nets with Inhibitors there is no more a sequentialization for every term.

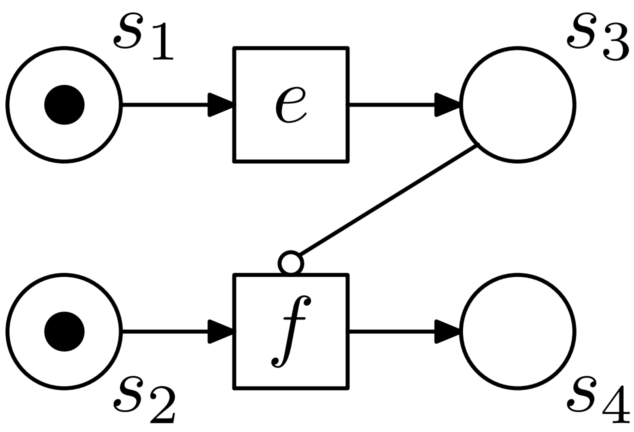


Only one valid (non-id) term, $e \oplus f$, no sequence.

The results - The difficult direction

We must find something like sequentialization.

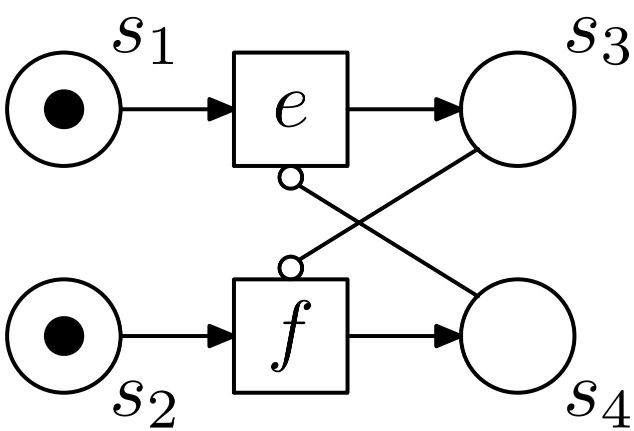
Axiom 7 - sequentialization of:



The results - The difficult direction

We must find something like sequentialization.

How do we sequentialize

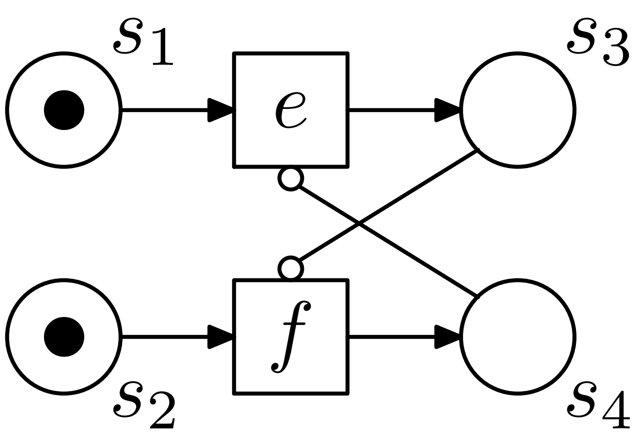


?

The results - The difficult direction

We must find something like sequentialization.

How do we sequentialize



?

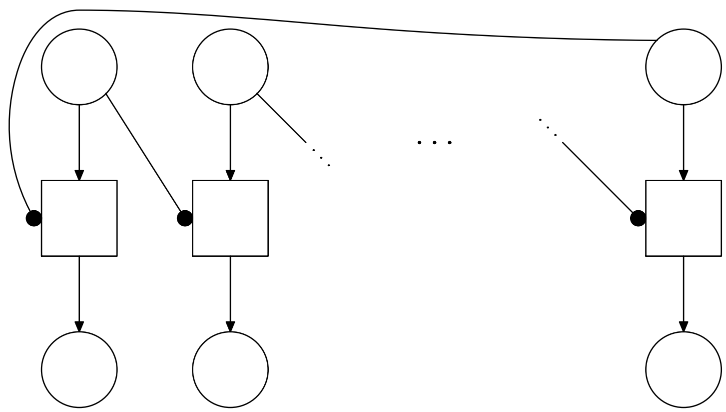
We do not. This is an "atomic" *generalized* transaction.

Generalized transitions

A generalized transition is

- A set of transitions $\{t_1, t_2, \ldots t_k\}$ such that $t_1 \sqsubset t_2,\ t_2 \sqsubset t_3, \ldots,\ t_k \sqsubset t_1$.

- An atomic transition t such that no set described above exists that would contain t



Modification of the proof

Now, the proof works fine again with

- modified sequentialization “procedure”
- $\oplus$ instead of $\|$
- generalized transitions in place of simple transitions

Conclusion

- The partial algebra approach can be adapted
  - to various net extensions
  - to various underlying *algebraic structures*

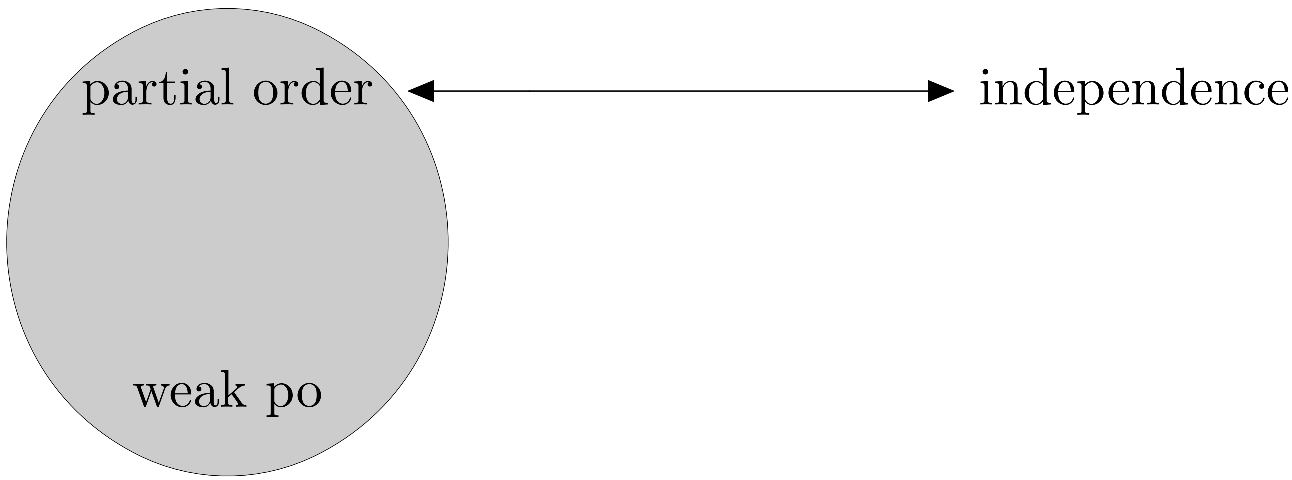
Philosophical point of view

Duality exists between independence and partial order.
Yet they are equivalent in a sense.

$$\text{partial order} \longleftrightarrow \text{independence}$$

Philosophical point of view

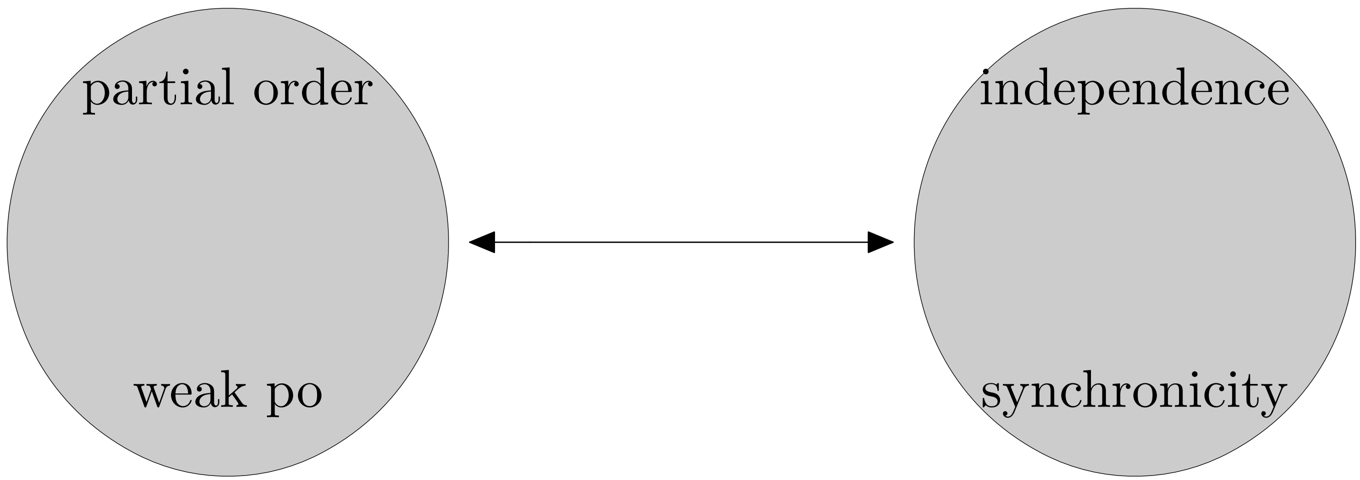
Duality exists between independence and partial order.
Yet they are equivalent in a sense.



We add to the causality side the “not later than” relation,

Conclusion from philosophical point of view

Duality exists between independence and partial order.
Yet they are equivalent in a sense.



We add to the causality side the “not later than” relation,
and add synchronicity on the right side to “keep it balanced”.

Bibliography

- Jörg Desel, Gabriel Juhás, and Robert Lorenz: Petri nets over partial algebra. In *Unifying Petri Nets*, number 2128 in LNCS. Springer, 2001.

- Jetty Kleijn and Maciej Koutny: Process semantics of P/T-Nets with inhibitor arcs. In *Proc. of 21th ICATPN*, number 1825 in LNCS. Springer, 2000.

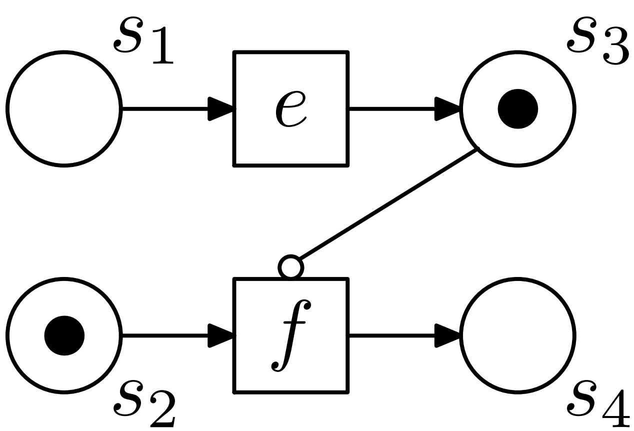
- José Meseguer and Ugo Montanari: Petri nets are monoids. *Information and Computation*, 88(2):105–155, October 1990.

Thank you!

Please feel free to ask a question.

This slide is intentionally left blank.

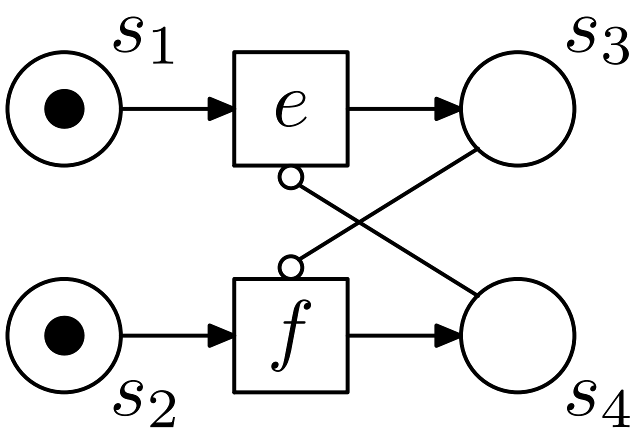
Nope!



f is disabled after firing e ! Not a very good description.

Back

Nope!



This would imply that $f \prec e \wedge e \prec f$. Not even a partial order!

Back