Discrete Parameters in Petri Nets

SYNCOP2015

Based on a Paper accepted in PN2015

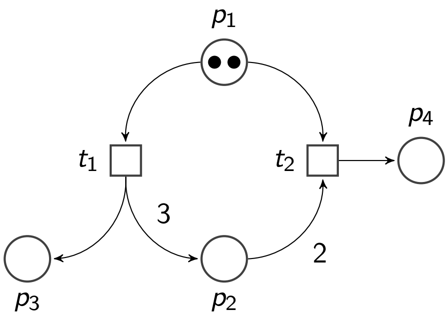
Nicolas David, Claude Jard, Didier Lime, Olivier H. Roux

April 22, 2015

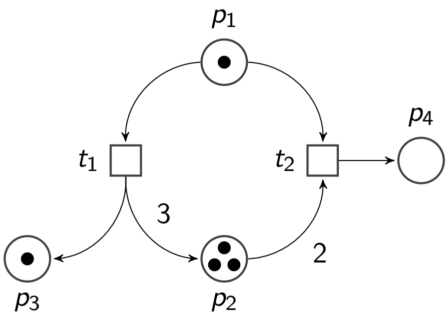
Why Introducing Parameters ?

- modeling arbitrary large amount of processes (markings)
- modeling unspecified aspect of the environement
- ...

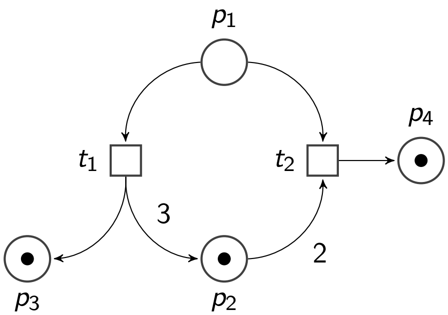
Classic Model ? a marked Petri Net (PPN)



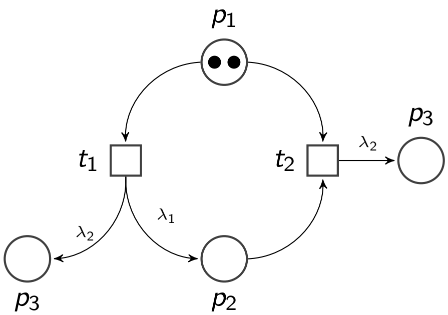
Classic Model ? a marked Petri Net (PPN)

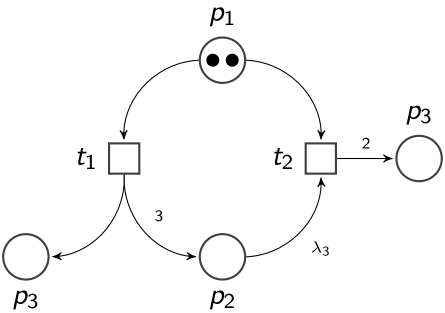


Classic Model ? a marked Petri Net (PPN)

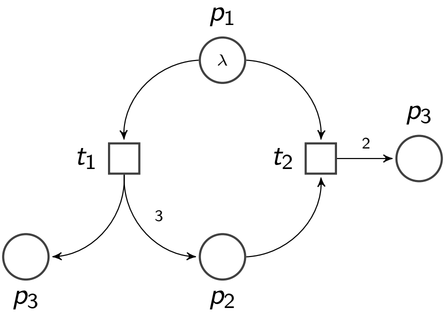


Generalization toward Parametric marked Petri Net (PPN)

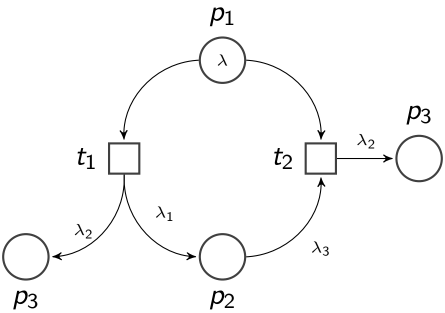



p_1
t_1
t_2
2
p_3
3
λ_3
p_3
p_2

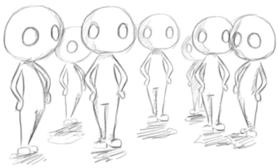
Generalization toward Parametric marked Petri Net (PPN)



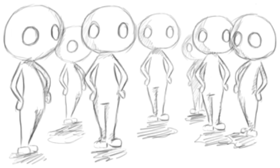
Generalization toward Parametric marked Petri Net (PPN)

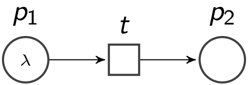


Some concrete examples



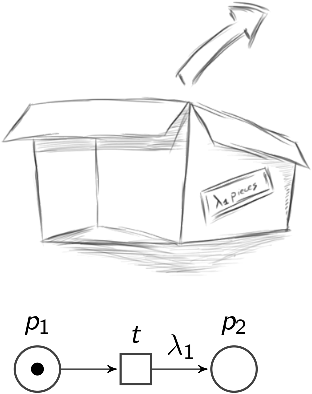
Some concrete examples

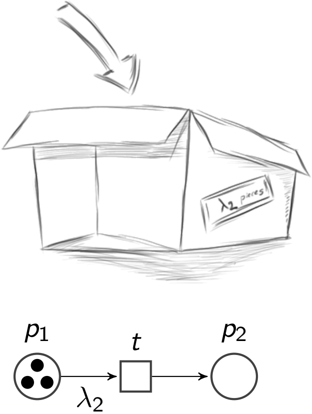




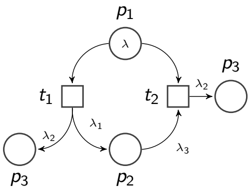
Some Concrete Examples

Some Concrete Examples

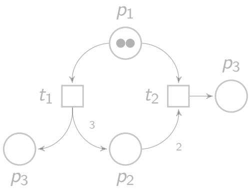




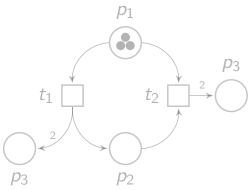
Instantiation



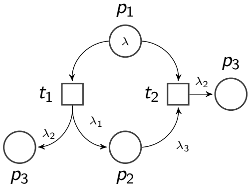
$$\begin{aligned}\nu_1(\lambda) &= 2\\ \nu_1(\lambda_1) &= 3\\ \nu_1(\lambda_2) &= 1\\ \nu_1(\lambda_3) &= 2\end{aligned}$$



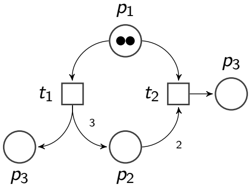
$$\begin{aligned}\nu_2(\lambda) &= 3\\ \nu_2(\lambda_1) &= 1\\ \nu_2(\lambda_2) &= 2\\ \nu_2(\lambda_3) &= 1\end{aligned}$$



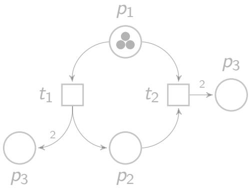
Instantiation



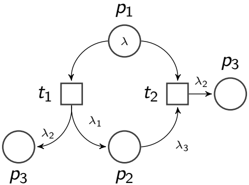
$$
\begin{aligned}
\nu_1(\lambda) &= 2\\
\nu_1(\lambda_1) &= 3\\
\nu_1(\lambda_2) &= 1\\
\nu_1(\lambda_3) &= 2
\end{aligned}
$$



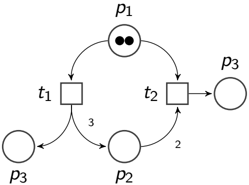
$$
\begin{aligned}
\nu_2(\lambda) &= 3\\
\nu_2(\lambda_1) &= 1\\
\nu_2(\lambda_2) &= 2\\
\nu_2(\lambda_3) &= 1
\end{aligned}
$$



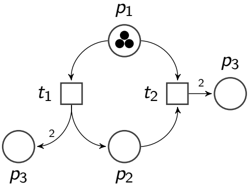
Instantiation



$$\begin{aligned} \nu_1(\lambda) &= 2 \\ \nu_1(\lambda_1) &= 3 \\ \nu_1(\lambda_2) &= 1 \\ \nu_1(\lambda_3) &= 2 \end{aligned}$$



$$\begin{aligned} \nu_2(\lambda) &= 3 \\ \nu_2(\lambda_1) &= 1 \\ \nu_2(\lambda_2) &= 2 \\ \nu_2(\lambda_3) &= 1 \end{aligned}$$



Reminders...

Definition (Reachability)

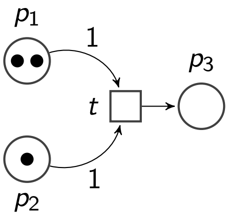
Let $\mathcal{S} = (\mathcal{N}, m_0) = (P, T, Pre, Post, m_0)$ *and* m *a marking of* $\mathcal{S}$, $\mathcal{S}$ *reaches* m *iff* $m \in RS(\mathcal{S})$.

Definition (Coverability)

Let $\mathcal{S} = (\mathcal{N}, m_0) = (P, T, Pre, Post, m_0)$ *and* m *a marking of* $\mathcal{S}$, $\mathcal{S}$ *covers* m *if there exists a reachable marking* m' *of* $\mathcal{S}$ *such that* m' *is greater or equal to* m *i.e.*

$$\exists m' \in \mathcal{RS}(\mathcal{S}) s.t.\ \forall p \in P, m'(p) \geq m(p) \tag{1}$$

Decidability studied in [2] and [1]



$$RS = \{(2,1,0),(1,0,1)\}$$

$$CS = \{m | m \leq (2,1,0) \vee m \leq (1,0,1)\}$$

Given a class of problem $\mathcal{P}$ (coverability, reachability,...), $\mathcal{SP}$ a PPN and ϕ is an instance of $\mathcal{P}$

Definition ($\mathcal{P}$-Existence problem)

($\mathscr{E}$-$\mathcal{P}$): Is there a valuation $\nu \in \mathbb{N}^{Par}$ s.t. $\nu(\mathcal{SP})$ satisfies ϕ ?

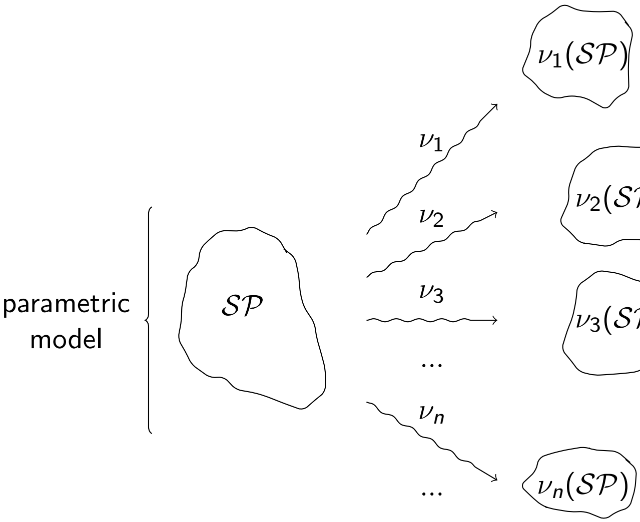
Definition ($\mathcal{P}$-Universality problem)

($\mathscr{U}$-$\mathcal{P}$): Does $\nu(\mathcal{SP})$ satisfies ϕ for each $\nu \in \mathbb{N}^{Par}$?

Definition ($\mathcal{P}$-Synthesis problem)

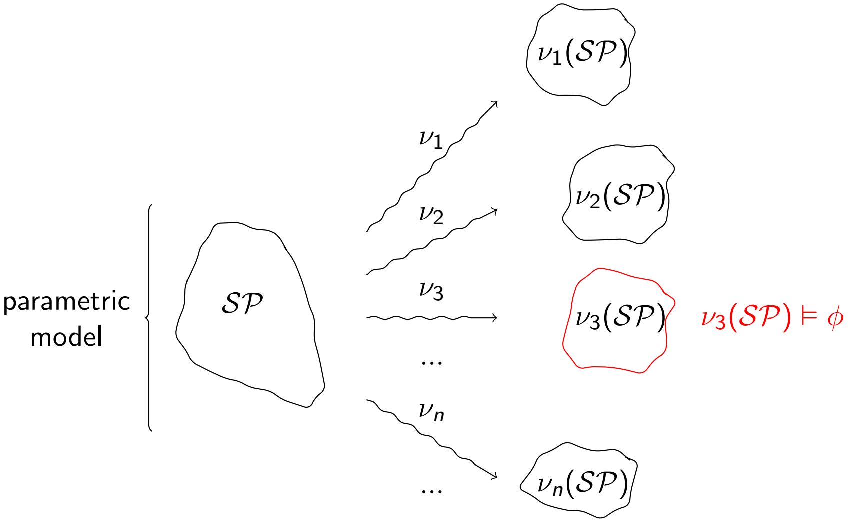
($\mathscr{S}$-$\mathcal{P}$): Give all the valuation ν, s.t. $\nu(\mathcal{SP})$ satisfies ϕ.

Existence

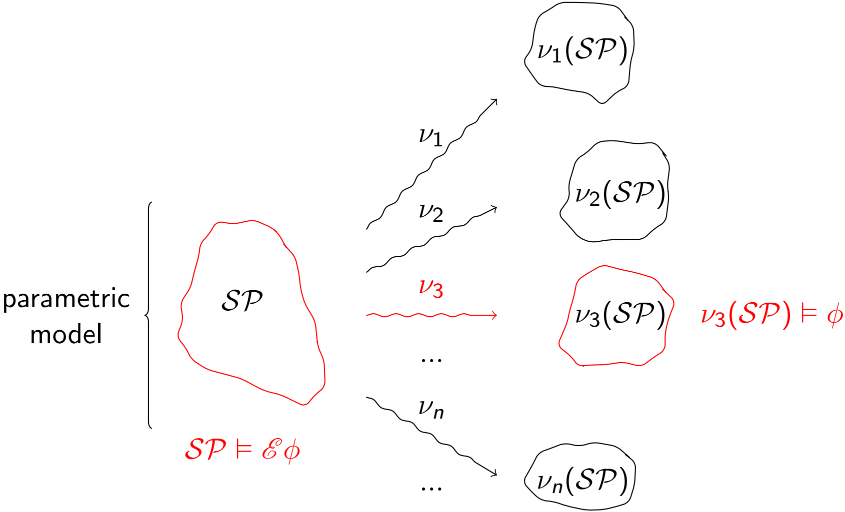


parametric model
$\mathcal{SP}$
ν_1
ν_2
ν_3
...
ν_n
...
$\nu_1(\mathcal{SP})$
$\nu_2(\mathcal{SP})$
$\nu_3(\mathcal{SP})$
$\nu_n(\mathcal{SP})$

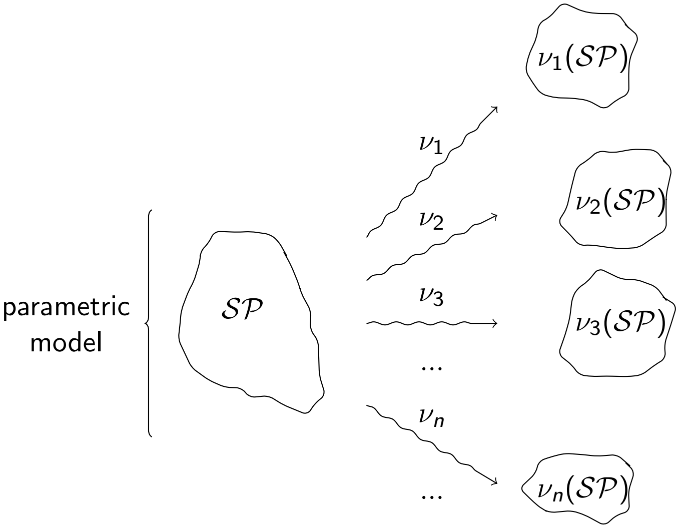
Existence


parametric model
$\mathcal{SP}$
ν_1
ν_2
ν_3
...
ν_n
...
$\nu_1(\mathcal{SP})$
$\nu_2(\mathcal{SP})$
$\nu_3(\mathcal{SP})$
$\nu_n(\mathcal{SP})$
$\nu_3(\mathcal{SP}) \models \phi$

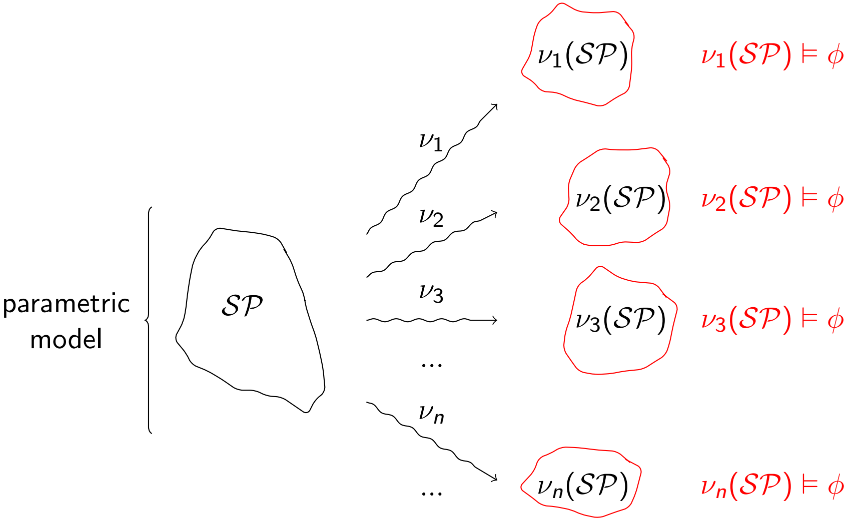
Existence


parametric
model
$\mathcal{SP}$
$\mathcal{SP} \models \mathscr{E}\phi$
ν_1
ν_2
ν_3
...
ν_n
...
$\nu_1(\mathcal{SP})$
$\nu_2(\mathcal{SP})$
$\nu_3(\mathcal{SP})$
$\nu_3(\mathcal{SP}) \models \phi$
$\nu_n(\mathcal{SP})$

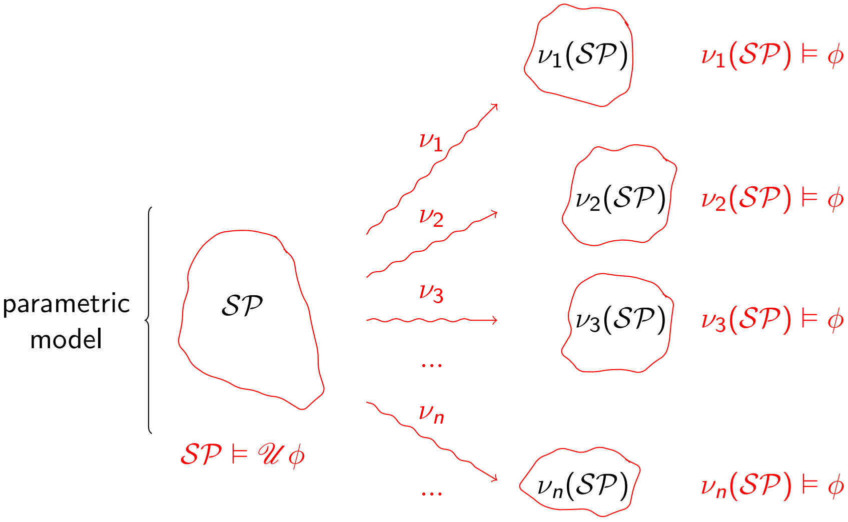
Universality



Universality



Universality



Mixing Properties and Parameters...

($\mathscr{U}$-cov) asks:

"Does each valuation of the parameters implies that the valuation of the PPN covers m ?"

i.e.

$$m \text{ is } \mathscr{U}\text{-coverable in } \mathcal{SP} \Leftrightarrow \left\{ \begin{array}{ll} \forall \nu \in \mathbb{N}^{Par}, & \exists m' \in \mathcal{RS}(\nu(\mathcal{SP})) \\ & \text{s.t. } m' \geq m \end{array} \right. \tag{2}$$

Results

Theorem (Undecidability of $\mathscr{E}$-cov on PPN)

The $\mathscr{E}$-coverability problem for PPN is undecidable.

Theorem (Undecidability of $\mathscr{U}$-cov on PPN)

The $\mathscr{U}$-coverability problem for PPN is undecidable.

2-Counters Machine

- two counters c_1, c_2,
- states $P = \{p_0, ...p_m\}$, a terminal state labelled *halt*
- finite list of instructions $I_1, ..., I_s$ among the following list:
  - increment a counter
  - decrement a counter
  - check if a counter equals zero

Counters are assumed positive.

Example of 2-Counters Machine

$$p_1.\ C_0 := C_0 + 1; goto\ p_2;$$

$$p_2.\ C_1 := C_1 + 1; goto\ p_1;$$

instructions sequence:
$(p_1, C_1 = 0, C_2 = 0)$
$\rightarrow (p_2, C_1 = 1, C_2 = 0)$
$\rightarrow (p_1, C_1 = 1, C_2 = 1)$
$\rightarrow (p_2, C_1 = 2, C_2 = 1)$
$\rightarrow ...$

Undecidability

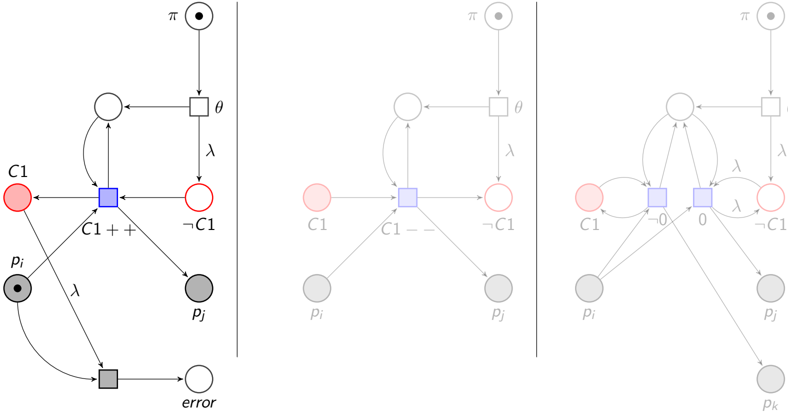
- *halting problem* (whether state *halt* is reachable) is *undecidable*
- *counters boundedness problem* (whether the counters values stay in a finite set) is *undecidable*

proved by Minksy [3]

Why ?

- simulation of a counter machine
- $\mathscr{E}$-cov can be reduced to *halting problem*
- $\mathscr{U}$-cov can be reduced to *counter boundedness*

Simulation of Instructions: $m(C1) + m(\neg C1) = \lambda$

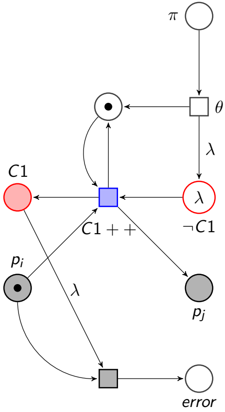


incrementation of a counter

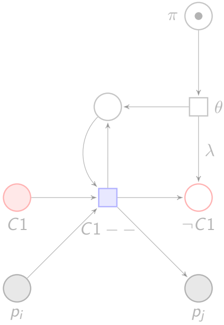
decrementation of a counter

zero test of a counter

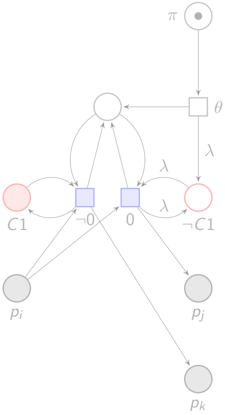
Simulation of Instructions: $m(C1) + m(\neg C1) = \lambda$



incrementation
of a counter

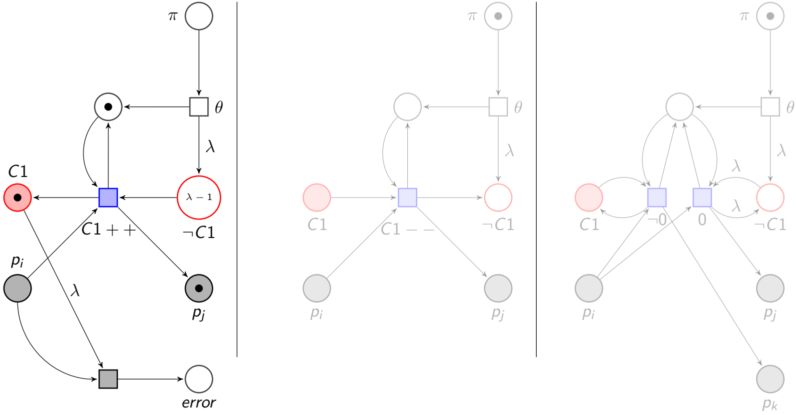


decrementation
of a counter



zero test of
a counter

Simulation of Instructions: $m(C1) + m(\neg C1) = \lambda$

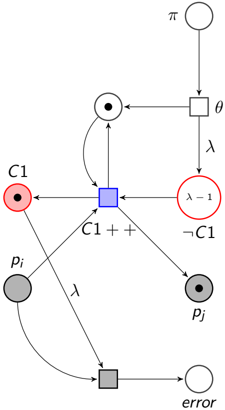


incrementation of a counter

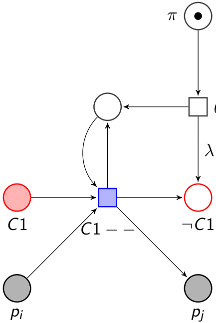
decrementation of a counter

zero test of a counter

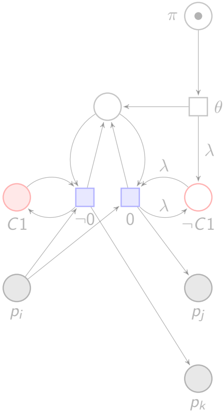
Simulation of Instructions: $m(C1) + m(\neg C1) = \lambda$



incrementation
of a counter

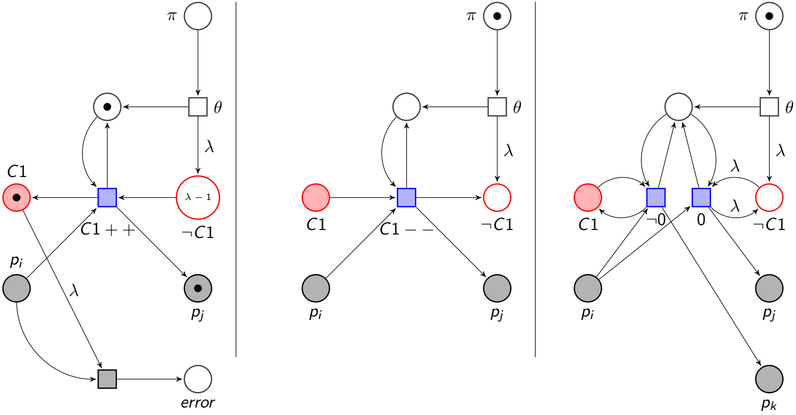


decrementation
of a counter



zero test of
a counter

Simulation of Instructions: $m(C1) + m(\neg C1) = \lambda$



incrementation of a counter

decrementation of a counter

zero test of a counter

- $\mathcal{M}$ halts iff there exists a valuation ν such that $\nu(\mathcal{SP}_{\mathcal{M}})$ covers the corresponding p_{halt} place.
- the counters are unbounded along the instructions sequence of $\mathcal{M}$ iff for each valuation ν, $\nu(\mathcal{SP}_{\mathcal{M}})$ covers the *error state*.

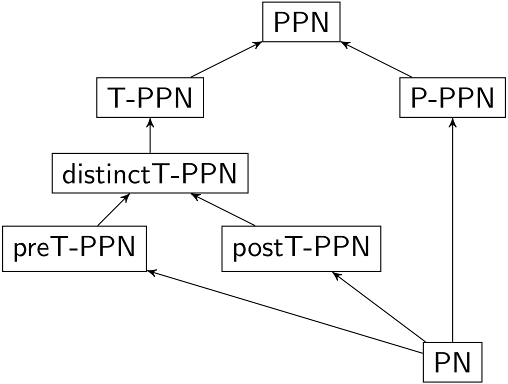
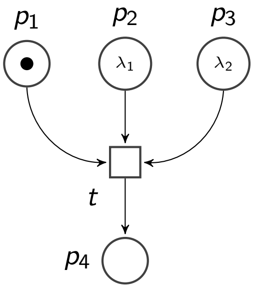
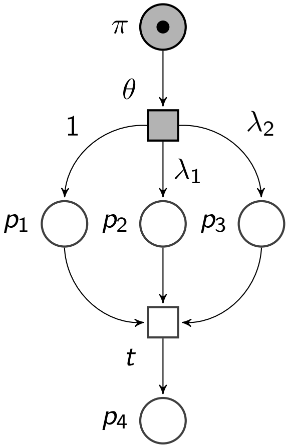


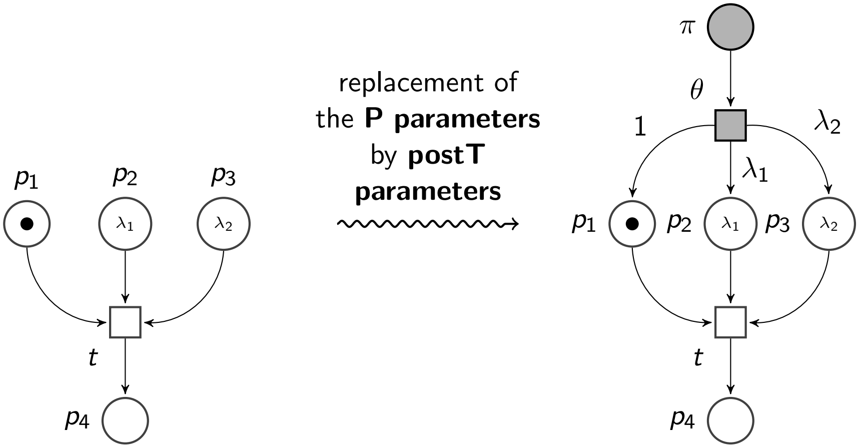
Figure 1: Syntaxical subclasses of PPN and inclusions between them



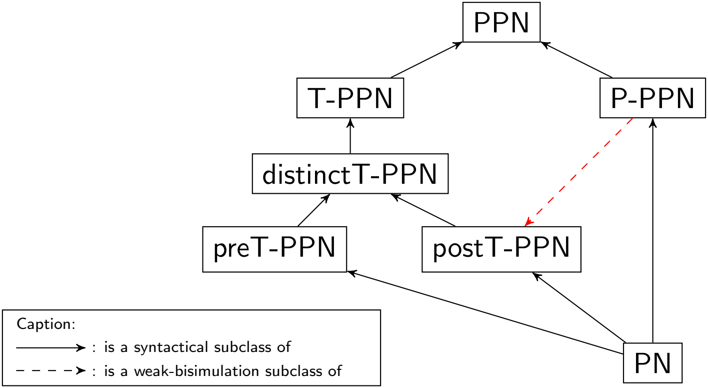
replacement of the **P parameters** by **postT parameters**



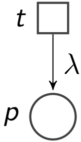
From P-PPN to postT-PPN



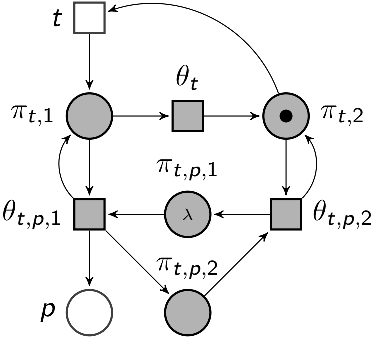
replacement of the **P parameters** by **postT parameters**


PPN
T-PPN
P-PPN
distinctT-PPN
preT-PPN
postT-PPN
PN
Caption:
: is a syntactical subclass of
: is a weak-bisimulation subclass of

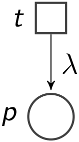
From postT-PPN to P-PPN



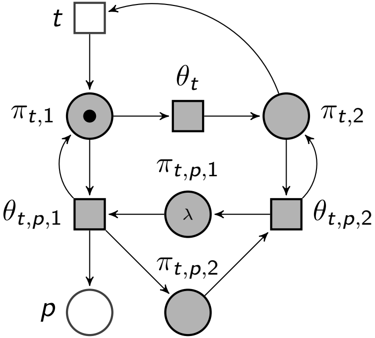
replacement
of the **postT parameters** by
P parameters



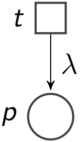
From postT-PPN to P-PPN



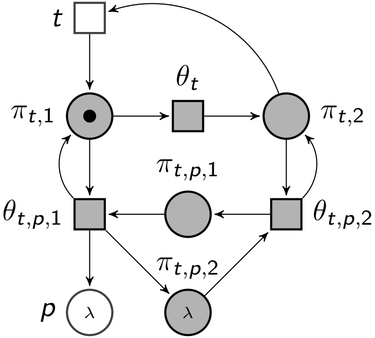
replacement
of the **postT**
parameters by
P parameters



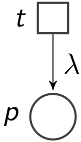
From postT-PPN to P-PPN



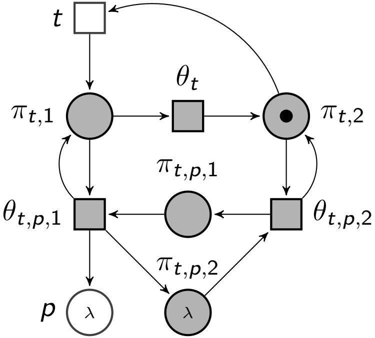
replacement
of the **postT**
parameters by
P parameters



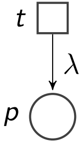
From postT-PPN to P-PPN



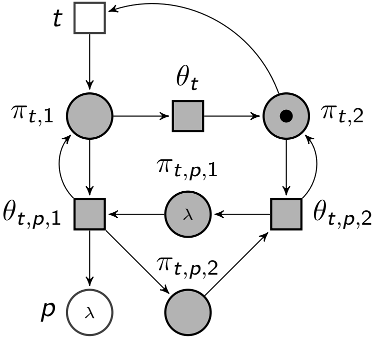
replacement
of the **postT**
parameters by
P parameters

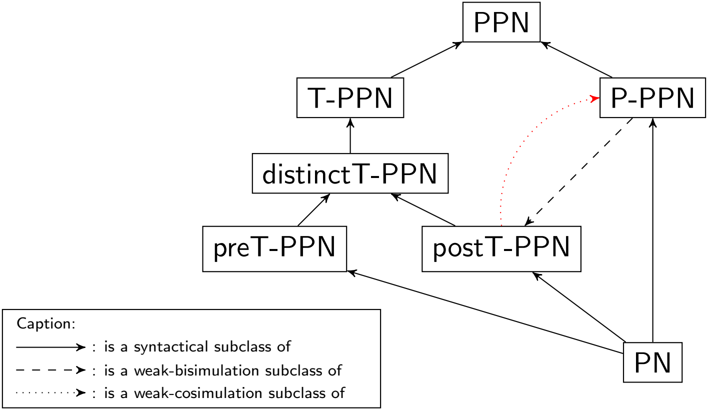


From postT-PPN to P-PPN



replacement
of the **postT**
parameters by
P parameters




PPN
T-PPN
P-PPN
distinctT-PPN
preT-PPN
postT-PPN
PN
Caption:
: is a syntactical subclass of
: is a weak-bisimulation subclass of
: is a weak-cosimulation subclass of

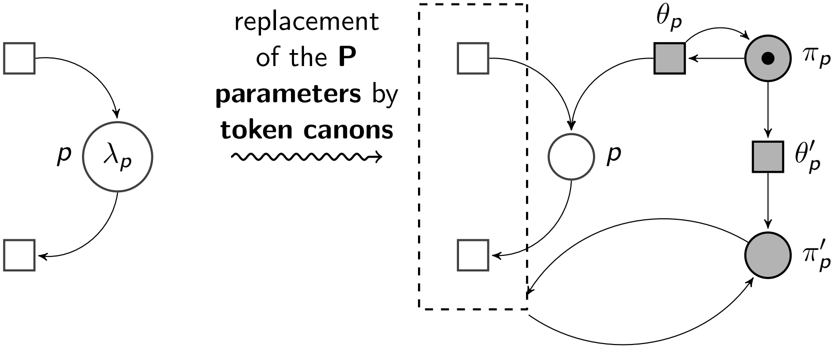


Figure 2: From PPN to PN

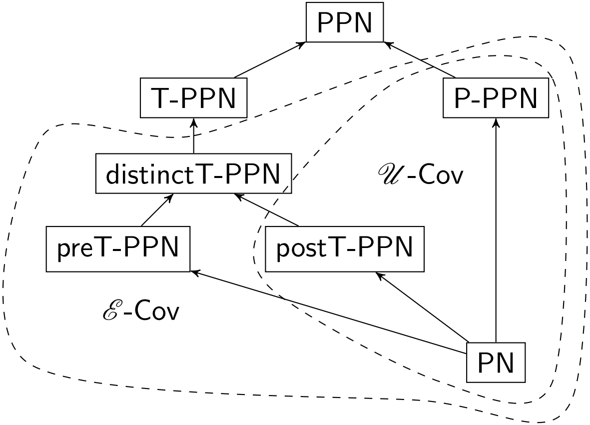


Figure 3: What is decidable among the subclasses ? *(for coverability)*

	$\mathscr{U}$-problem		$\mathscr{E}$-problem	
	Reach.	Cov.	Reach.	Cov.
preT-PPN	?	?	?	D
postT-PPN	?	D	?	D
PPN	U	U	U	U
distinctT-PPN	?	?	?	D
P-PPN	?	D	D	D

Table 1: Decidability results for parametric coverability and reachability

Future Work

- Fill the Table (especially $\mathscr{U}$-cov on preT-PPN...)
- Synthesis Problem

Richard M. Karp and Raymond E. Miller.
Parallel program schemata.
Journal of Computer and System Sciences, 3(2):147 – 195, 1969.

Ernst W. Mayr.
An algorithm for the general petri net reachability problem.
In *Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computing*, STOC '81, pages 238–246, New York, NY, USA, 1981. ACM.

Marvin L. Minsky.
Computation: Finite and Infinite Machines.
Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1967.