Logics for Petri nets with propagating failures

Leandro Gomes, Alexandre Madeira and Mario Benevides

8^{th} IPM International Conference on Fundamentals of Software Engineering, 2^{nd} May 2019

Outline

Outline

Propositional dynamic logic (in a rush)

Signatures Are pairs (Prop, Π) where Prop and Π are disjoint sets of **propositions**, and atomic programs

Propositional dynamic logic (in a rush)

Signatures Are pairs (Prop, Π) where Prop and Π are disjoint sets of **propositions**, and atomic programs

Sentences $\varphi ::= \mathbf{p} \mid \langle\pi\rangle\varphi \mid [\pi]\varphi \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$

$\pi ::= \pi_0 \mid \pi + \pi \mid \pi;\pi \mid \pi^* \mid ?\varphi$

and $p \in \mathrm{Prop}$

Propositional dynamic logic (in a rush)

Signatures Are pairs (Prop, Π) where Prop and Π are disjoint sets of **propositions**, and atomic programs

Sentences $\varphi ::= \mathbf{p} \mid \langle\pi\rangle\varphi \mid [\pi]\varphi \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi$

$\pi ::= \pi_0 \mid \pi + \pi \mid \pi;\pi \mid \pi^* \mid ?\varphi$

and $p \in \mathrm{Prop}$

Models Models are Kripke structures, i.e. tuples (W, V, R)

W is a set

$V : \mathrm{Prop} \to \mathcal{P}(W)$ is a function

$R = (R_\pi \subseteq W \times W)_{\pi \in \Pi}$ is an Π-family of binary relations

Propositional dynamic logic (in a rush)

Satisfaction

$M, w \models p$ iff $w \in V(p)$

$M, w \models \varphi \wedge \varphi'$ iff $M, w \models \varphi$ and $M, w \models \varphi'$

$M, w \models \varphi \vee \varphi'$ iff $M, w \models \varphi$ or $M, w \models \varphi'$

$M, w \models \neg\varphi$ iff it is false that $M, w \models \varphi$

Propositional dynamic logic (in a rush)

Satisfaction

$M, w \models p$ iff $w \in V(p)$

$M, w \models \varphi \wedge \varphi'$ iff $M, w \models \varphi$ and $M, w \models \varphi'$

$M, w \models \varphi \vee \varphi'$ iff $M, w \models \varphi$ or $M, w \models \varphi'$

$M, w \models \neg\varphi$ iff it is false that $M, w \models \varphi$

$M, w \models \langle\pi\rangle\varphi$ **iff there is a** $w' \in W$ **such that** $(w, w') \in R_a$ **and** $M, w' \models \varphi$**;**

$M, w \models [\pi]\varphi$ **iff for any** $w' \in W$ **such that** $(w, w') \in R_a$ **we have** $M, w' \models \varphi$**;**

Petri Nets

- It is a tuple of the form $(\mathcal{P}, \mathcal{T}, w, m_0)$ where:
  - $\mathcal{P}$ represents a finite set of *places*;
  - $\mathcal{T}$ represents a finite set of *transitions*;
  - w: $(\mathcal{P} \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P}) \rightarrow \mathbb{N}_0$ attributes to each arc a non-negative integer, representing its multiplicity.
  - m_0: $\mathcal{P} \rightarrow \mathbb{N}_0$ defines an *Initial Marking*

Example

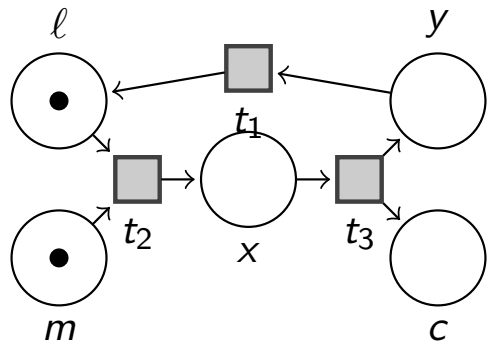


Figure: Example of a Petri net

Proposition dynamic logic for Petri nets: Petri-PDL*

(Bruno Lopes, 2014) and (Mario Benevides, 2016)

Proposition dynamic logic for Petri nets: Petri-PDL*

(Bruno Lopes, 2014) and (Mario Benevides, 2016)

Basic programs $\Pi_0(P)$:

$$\pi ::= at_1b \mid abt_2c \mid at_3bc$$

where t_i is of type $T_i, i = 1, 2, 3$ and $a, b, c \in P$

Proposition dynamic logic for Petri nets: Petri-PDL*

(Bruno Lopes, 2014) and (Mario Benevides, 2016)

Basic programs $\Pi_0(P)$:

$$\pi ::= at_1b \mid abt_2c \mid at_3bc$$

where t_i is of type $T_i, i = 1, 2, 3$ and $a, b, c \in P$

Petri net Programs $\Pi(P)$:

$$\eta ::= \pi \mid \pi \odot \eta \mid \eta^\star$$

for $\pi \in \Pi_0(P)$

Proposition dynamic logic for Petri nets: Petri-PDL*

(Bruno Lopes, 2014) and (Mario Benevides, 2016)

Basic programs $\Pi_0(P)$:

$$\pi ::= at_1 b \mid abt_2 c \mid at_3 bc$$

where t_i is of type $T_i, i = 1, 2, 3$ and $a, b, c \in P$

Petri net Programs $\Pi(P)$:

$$\eta ::= \pi \mid \pi \odot \eta \mid \eta^\star$$

for $\pi \in \Pi_0(P)$

Formulas $\mathrm{Fm}^{\mathrm{Petri-PDL}^\star}(\mathrm{Prop})$:

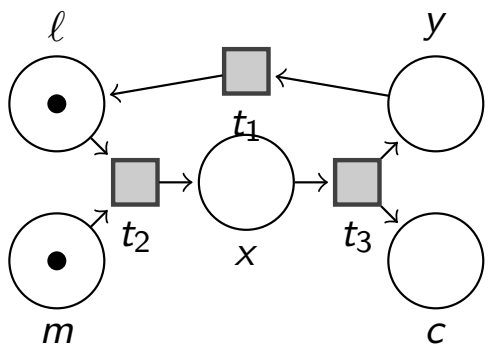
$$\rho ::= p \mid \top \mid \neg\rho \mid \rho \wedge \rho \mid \langle s, \eta \rangle \rho$$

where $p \in \mathrm{Prop}$.

Petri-PDL* dynamics – *firing* function

$$f : S \times \Pi_0 \to S$$

$$f(s, abt_2c) = \left\{ \begin{array}{rll} s_2 \mid c \in s_2, & \textit{if} & a, b \in s \\ \epsilon, & \textit{if} & a \notin s \ \textit{or} \ b \notin s \end{array} \right\}$$



Petri-PDL* dynamics – *firing* function

$$f : S \times \Pi_0 \to S$$

$$f(s, abt_2c) = \left\{ \begin{array}{rll} s_2 \mid c \in s_2, & \textit{if} & a, b \in s \\ \epsilon, & \textit{if} & a \notin s \ \textit{or} \ b \notin s \end{array} \right\}$$

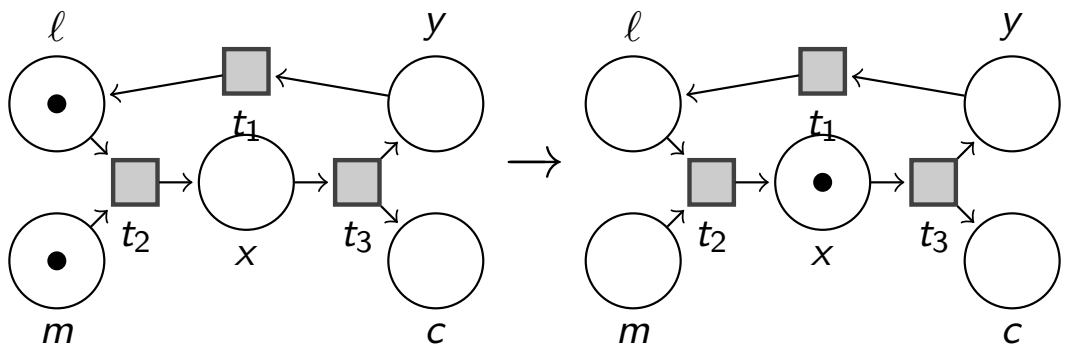


Figure: Dynamics for transitions of type abt_2c

Models:

$$\langle (\mathcal{W}, \mathcal{R}_\pi, \mathcal{M}), \mathbf{V} \rangle$$

- W is a non-empty set of states, $M : W \to S$, R_π is a binary relation over W.
- $\mathbf{V}$ is a valuation function $\mathbf{V}$: $\mathrm{Prop} \to 2^W$.

Models:

$$\langle (\mathcal{W}, \mathcal{R}_\pi, \mathcal{M}), \mathbf{V} \rangle$$

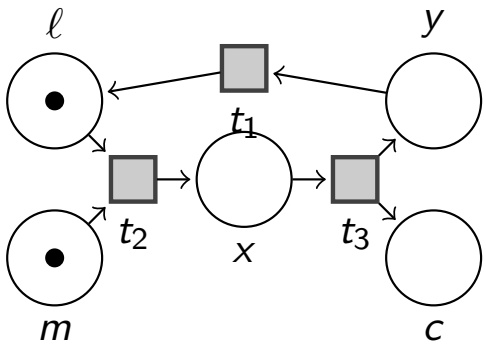
- W is a non-empty set of states, $M : W \to S$, R_π is a binary relation over W.
- $\mathbf{V}$ is a valuation function $\mathbf{V}: \mathrm{Prop} \to 2^W$.

Satisfaction:

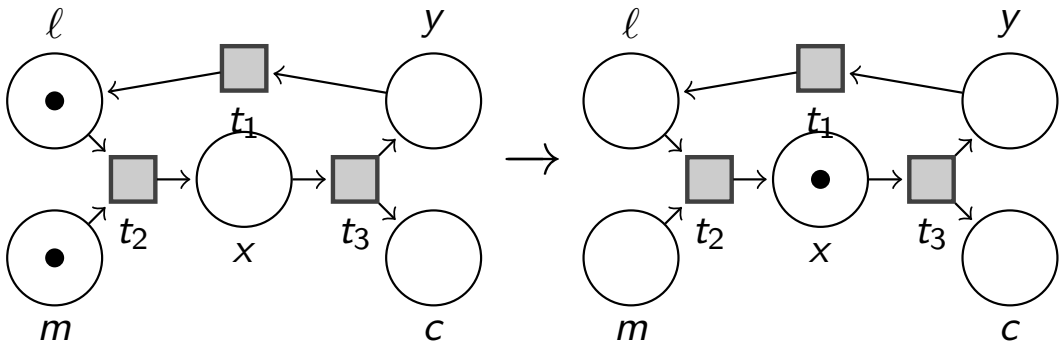
$$\mathbf{V}: \mathrm{Prop} \to 2^W$$

- $\mathcal{M}, w \models p$ iff $w \in \mathbf{V}(p)$;
- $\mathcal{M}, w \models \top$ always;
- $\mathcal{M}, w \models \neg\rho$ iff $\mathcal{M}, w \not\models \rho$;
- $\mathcal{M}, w \models \rho \wedge \rho'$ iff $\mathcal{M}, w \models \rho$ and $\mathcal{M}, w \models \rho'$;
- $\mathcal{M}, w \models \langle s, \eta \rangle \rho$ **if there exists** $w' \in W$, $wR_\eta w'$, $s \preceq M(w)$ **and** $\mathcal{M}, w' \models \rho$.

Example



Example



Example

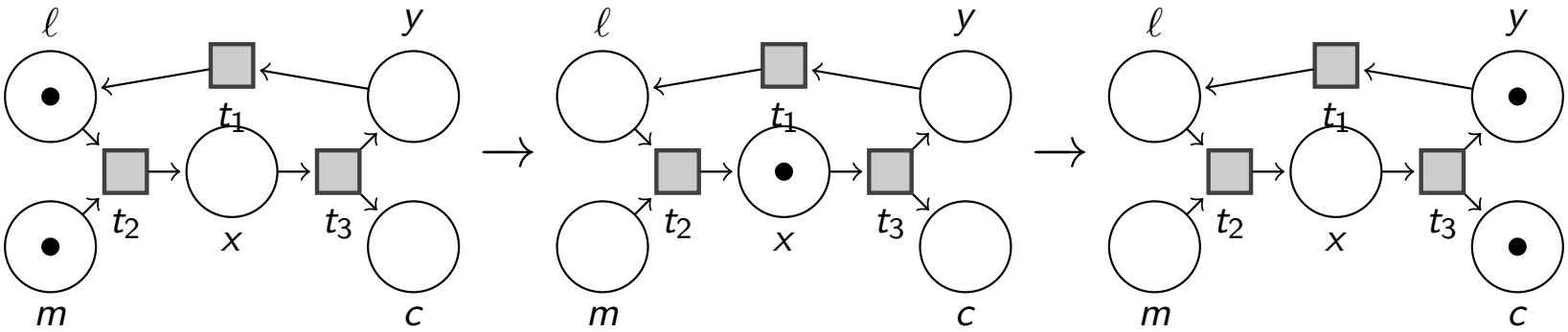


Figure: Execution of the program $\ell m t_2 x; x t_3 y c$ in a chocolate vending machine

$$\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c \rangle \top = \top$$

Failures in machines

After putting a coin in a vending machine, the desired chocolate gets stuck behind of the glass

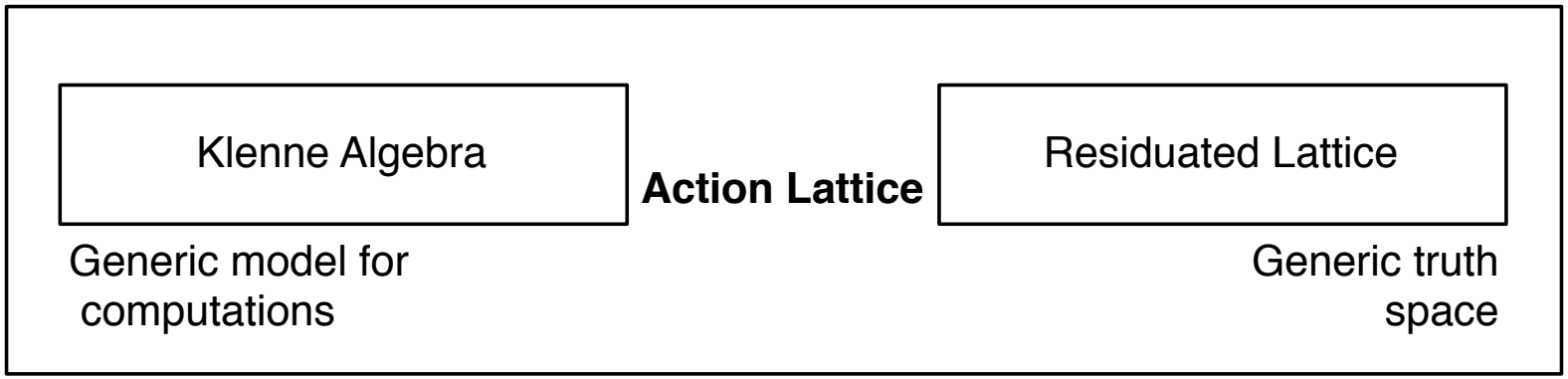
Failures in machines

After putting a coin in a vending machine, the desired chocolate gets stuck behind of the glass

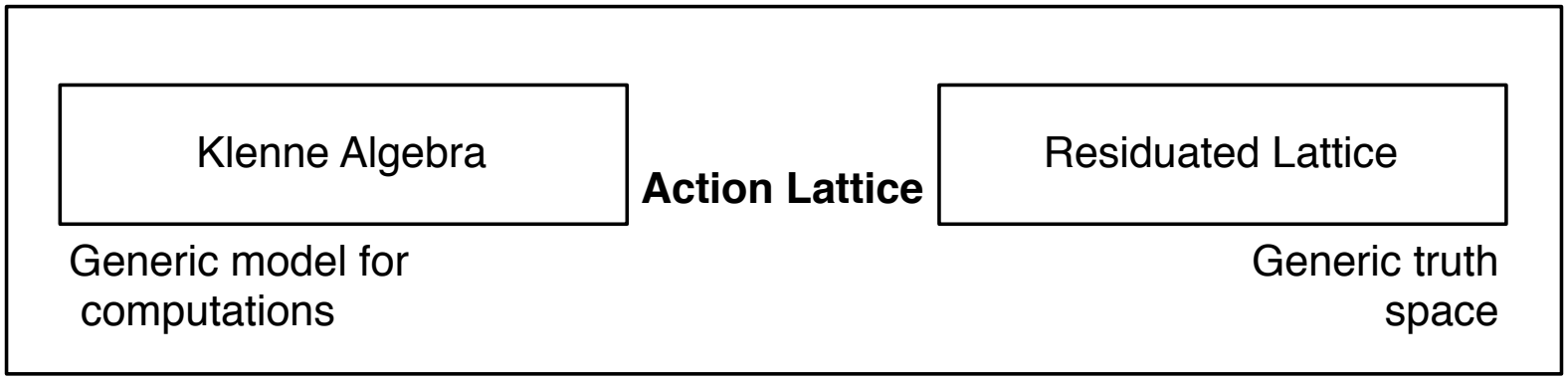
How to model these phenomena?

Outline

Construction parameter



Construction parameter



Action lattice (Pratt 90, Kozen 91)

$$A = (A, +, ;, 0, 1, *, \rightarrow, \cdot)$$

$(A, +, ;, 0, 1, *)$ is a **Kleene algebra**;

$\rightarrow$ is a **residue wrt** ;

$(A, +, .)$ is a **lattice** wrt relation $a \leq b \equiv a + b = b$

Examples: **2** - linear two-values lattice.

$$\mathbf{2} = (\{\top, \bot\}, \vee, \wedge, \bot, \top, *, \rightarrow, \wedge)$$

$\vee$	$\bot$	$\top$
$\bot$	$\bot$	$\top$
$\top$	$\top$	$\top$

$\wedge$	$\bot$	$\top$
$\bot$	$\bot$	$\bot$
$\top$	$\bot$	$\top$

$\rightarrow$	$\bot$	$\top$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\top$

$*$	
$\bot$	$\top$
$\top$	$\top$

Examples: $\mathbf{W}_k$ finite Wajsberg hoops

For a fixed natural $k > 0$ and a generator a,

$$\mathbf{W}_k = (W_k, +, ;, 0, 1, *, \to, \cdot)$$

where

- $W_k = \{a^0, a^1, \cdots, a^k\}$, $1 = a^0$ and $0 = a^k$,
- For any $m, n \leq k$:

$$a^m + a^n = a^{min\{m,n\}},$$
$$a^m ; a^n = a^{min\{m+n,k\}},$$
$$(a^m)^* = a^0,$$
$$a^m \to a^n = a^{max\{n-m,0\}},$$
$$a^m \cdot a^n = a^{max\{m,n\}}$$

Examples: Ł - the Łukasiewicz arithmetic lattice

$$\text{Ł} = ([0,1], \max, \odot, 0, 1, *, \rightarrow, \min)$$

where

$x \odot y = max\{0, y + x - 1\}$,

$x \rightarrow y = min\{1, 1 - x + y\}$ and

$*$ maps each point of $[0,1]$ to 1.

Petri net with **A**-failures

$$\mathcal{P} = (P, S, \Pi_0, I, M_0)$$

P is a set of places,

$S \subseteq P$ is the *set of (admissible) markings*,

$\Pi_0 \subseteq \Pi(P)$ is the *set of atomic programs*,

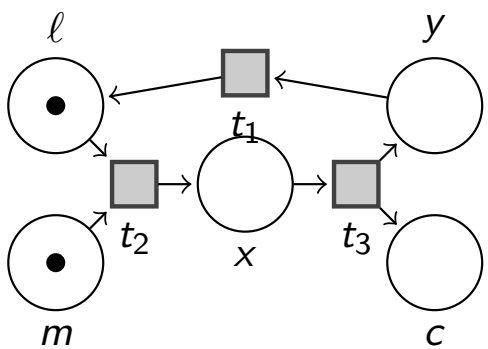
$I : \Pi_0 \to A$ is the *atomic programs reliability degree* and

$M_0 \in S$ is the *initial marking*.

Dynamics of Petri nets with **A**-failures - firing function

$$f_{\pi}^{I(\pi)} : S \times S \to A$$

$$f^{\alpha}_{abt_2c}(s, s') = \left\{ \begin{array}{r} \alpha, \text{ if } a, b \in s \text{ and } c \in s' \\ \alpha \to 0, \text{ if } a, b \in s \text{ and } a, b \in s' \\ 0 \text{ if } a, b \notin s \end{array} \right\}$$



Dynamics of Petri nets with **A**-failures - firing function

$$f_{\pi}^{I(\pi)} : S \times S \to A$$

$$f^{\alpha}_{abt_2c}(s, s') = \left\{ \begin{array}{r} \alpha, \text{ if } a, b \in s \text{ and } c \in s' \\ \alpha \to 0, \text{ if } a, b \in s \text{ and } a, b \in s' \\ 0 \text{ if } a, b \notin s \end{array} \right\}$$

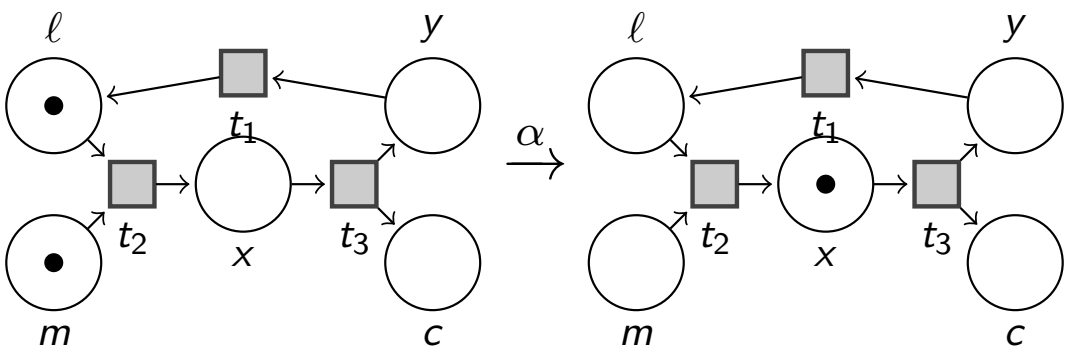


Figure: Dynamics of transition of type abt_2c

Interpretation of Petri net programs

The algebra of A-firing functions

$$\mathbf{F} = (F, \cup, \circ, \varnothing, \chi, *)$$

where:

F is the universe of all the α-firing functions, for all $\alpha \in A$

Interpretation of Petri net programs

The algebra of A-firing functions

$$\mathbf{F} = (F, \cup, \circ, \varnothing, \chi, *)$$

where:

F is the universe of all the α-firing functions, for all $\alpha \in A$

Theorem

F *is a Kleene Algebra*

Proof.

See (Gomes, 2017). □

Interpretation of Petri net programs

The algebra of A-firing functions

$$\mathbf{F} = (F, \cup, \circ, \varnothing, \chi, *)$$

where:

F is the universe of all the α-firing functions, for all $\alpha \in A$

Theorem

F *is a Kleene Algebra*

Proof.

See (Gomes, 2017). □

A-interpretations of programs

- for any atomic program $\pi \in \Pi_0$, $[\![\pi]\!](s, s') = f_\pi^{F(\pi)}(s, s')$
- $[\![\pi_1; \pi_2]\!](s, s') = ([\![\pi_1]\!] \circ [\![\pi_2]\!])(s, s')$
- $[\![\eta^*]\!](s, s') = [\![\eta]\!]^*(s, s')$, for $[\![\eta]\!]^*(s, s') = \sum_{i \geq 0} [\![\eta]\!]^i(s, s')$ where

  $[\![\eta]\!]^0(s, s') = \chi(s, s')$ and
  for any $i \geq 0$, $[\![\eta]\!]^{i+1}(s, s') = \big([\![\eta]\!]^i \circ [\![\eta]\!]\big)(s, s')$

Outline

PDL for Petri Nets with **A**-failures – **GP**(**A**)

The method is based on the one introduced in [Madeira, 2016].

PDL for Petri Nets with **A**-failures – **GP(A)**

The method is based on the one introduced in [Madeira, 2016].

Formulas:

The set of formulas $\mathrm{Fm}^{\mathrm{GP}(\mathbf{A})}(\mathrm{Prop}, P)$:

$$\rho ::= p \mid \top \mid \bot \mid \rho \wedge \rho \mid \rho \vee \rho \mid \rho \rightarrow \rho \mid \langle \eta \rangle \rho \mid [\eta]\rho$$

where $p \in \mathrm{Prop}$, $\eta ::= \pi \mid \pi; \eta \mid \eta^\star$ for $\pi ::= at_1 b \mid abt_2 c \mid at_3 bc$, and $a, b, c \in P$.

PDL for Petri Nets with **A**-failures – **GP(A)**

The method is based on the one introduced in [Madeira, 2016].

Formulas:

The set of formulas $\mathrm{Fm}^{\mathrm{GP}(\mathbf{A})}(\mathrm{Prop}, P)$:

$$\rho ::= p \mid \top \mid \bot \mid \rho \wedge \rho \mid \rho \vee \rho \mid \rho \rightarrow \rho \mid \langle \eta \rangle \rho \mid [\eta]\rho$$

where $p \in \mathrm{Prop}$, $\eta ::= \pi \mid \pi; \eta \mid \eta^{\star}$ for $\pi ::= at_1 b \mid abt_2 c \mid at_3 bc$, and $a, b, c \in P$.

Models:

$$\mathcal{M} = \langle \mathcal{P}, \mathbf{V} \rangle$$

$\mathcal{P}$ is a Petri net with **A**-failures and

V is a valuation function $\mathbf{V} : \mathrm{Prop} \times S \rightarrow A$.

PDL for Petri Nets with **A**-failures – **GP(A)**

(Graded) Satisfaction:

$$\models : (\mathcal{M} \times S) \times \mathrm{Fm}^{\mathrm{GP}(\mathbf{A})}(\mathrm{Prop}) \to A$$

- $(\mathcal{M}, s \models p) = \mathbf{V}(p, s)$;
- $(\mathcal{M}, s \models \top) = \top$;
- $(\mathcal{M}, s \models \bot) = \bot$;
- $(\mathcal{M}, s \models \rho \wedge \rho') = (\mathcal{M}, s \models \rho) \cdot (\mathcal{M}, s \models \rho')$;
- $(\mathcal{M}, s \models \rho \vee \rho') = (\mathcal{M}, s \models \rho) + (\mathcal{M}, s \models \rho')$;
- $(\mathcal{M}, s \models \rho \to \rho') = (\mathcal{M}, s \models \rho) \to (\mathcal{M}, s \models \rho')$;
- $(\mathcal{M}, s \models \langle \eta \rangle \rho) = \sum_{s' \in S} \Big([\![\eta]\!](s, s'); (\mathcal{M}, s' \models \rho) \Big)$;
- $(\mathcal{M}, s \models [\eta] \rho) = \bigwedge_{s' \in S} \Big([\![\eta]\!](s, s') \to (\mathcal{M}, s' \models \rho) \Big)$

Example

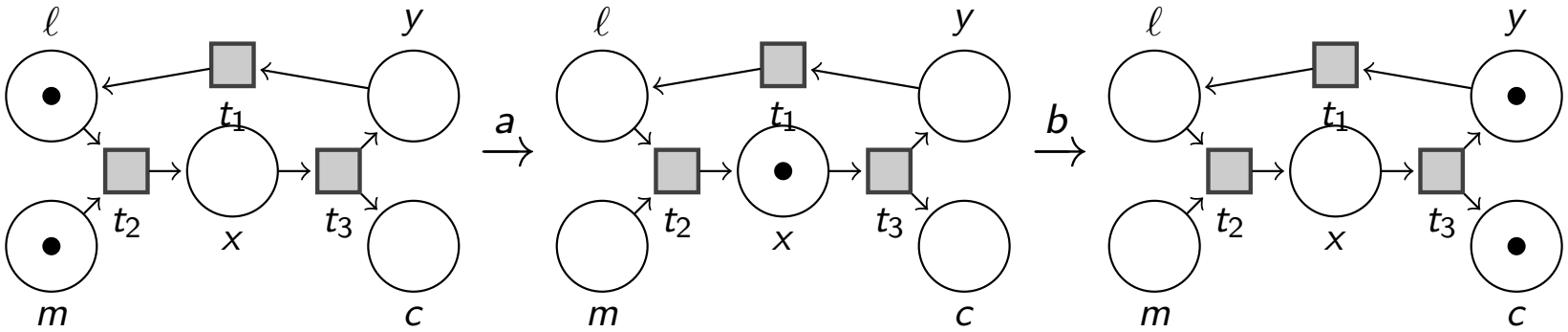


Figure: A Petri net for a (possibly) defective chocolate vending machine

Example

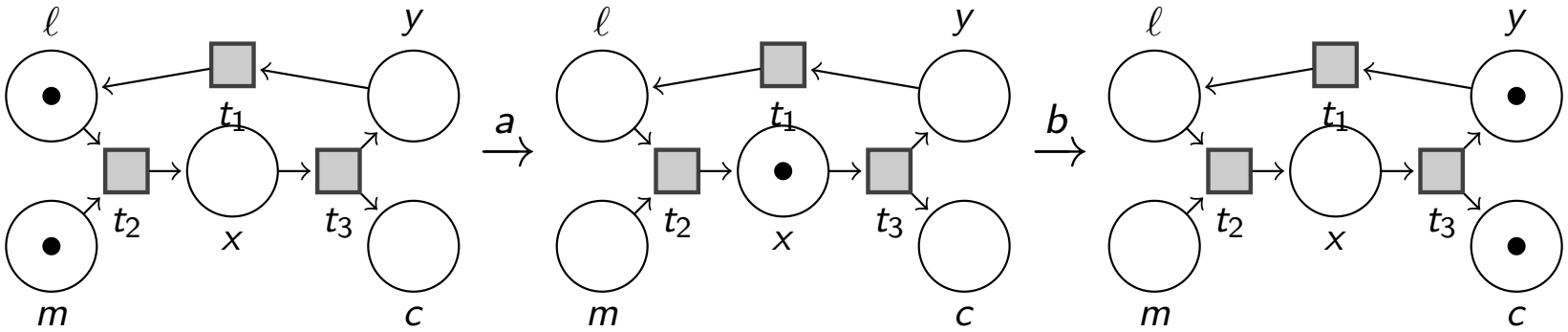


Figure: A Petri net for a (possibly) defective chocolate vending machine

GP(2)

Example

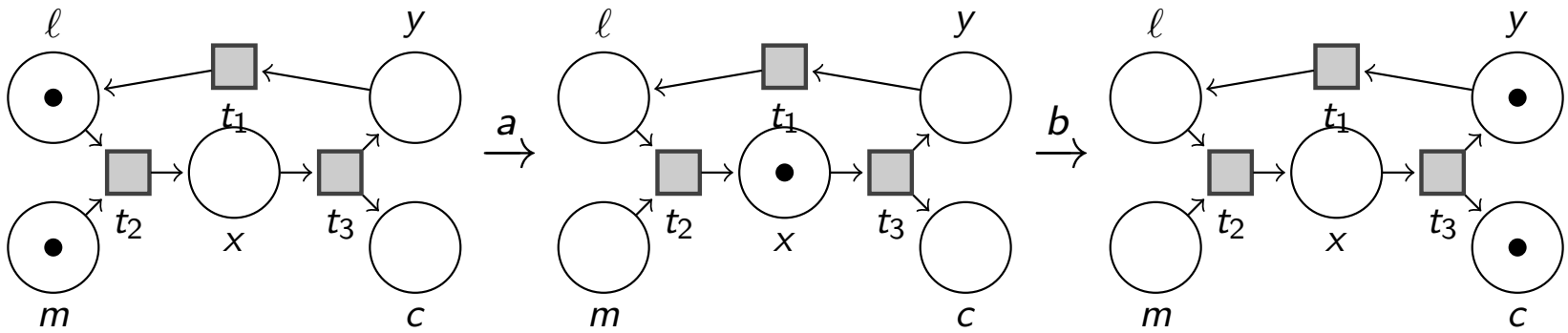


Figure: A Petri net for a (possibly) defective chocolate vending machine

GP(2)

$a = \top,\ b = \top$

$\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c \rangle \top =$

$= \sum_{s \in S} \Big(\big([\![\ell m t_2 x]\!](\ell m, x); [\![x t_3 y c]\!](x, s) + [\![\ell m t_2 x]\!](\ell m, \ell m); [\![x t_3 y c]\!](\ell m, s) \big); (M, s \models \top) \Big)$

$= \top$

$\mathbf{GP}(W_{10})$

$\mathbf{GP}(W_{10})$

$a = 8,\ b = 9$

$$\begin{aligned}
&\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c \rangle \top \ = \\
&= \big(8; 9 + (8 \to 0); 0\big); 10 + \big((8 \to 0); 0\big); 10 + \big(8; (9 \to 0) + (8 \to 0); 0\big); 10 \\
&= \ 7.
\end{aligned}$$

$\mathbf{GP}(W_{10})$

$a = 8,\ b = 9$

$$\begin{aligned}
&\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c \rangle \top \ = \\
&= \big(8; 9 + (8 \to 0); 0\big); 10 + \big((8 \to 0); 0\big); 10 + \big(8; (9 \to 0) + (8 \to 0); 0\big); 10 \\
&= \ 7.
\end{aligned}$$

$\mathbf{GP}(Ł)$

$\mathbf{GP}(W_{10})$

$a = 8,\ b = 9$

$$\begin{aligned}&\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c\rangle \top\ = \\ =&\big(8; 9 + (8 \to 0); 0\big); 10 + \big((8 \to 0); 0\big); 10 + \big(8; (9 \to 0) + (8 \to 0); 0\big); 10 \\ =&\ 7.\end{aligned}$$

$\mathbf{GP}(Ł)$

$a = 0.78,\ b = 0.93$

$$\begin{aligned}&\mathcal{M}, \ell m \models \langle \ell m t_2 x; x t_3 y c\rangle \top = \\ =&max\{max\{0.78 \odot 0.93, 0\} \odot 1, 0, max\{0.78 \odot (0.93 \to 0), 0\} \odot 1\} \\ =&0.71\end{aligned}$$

Outline

Conclusions

We achieved

Generalisation of Petri PDL*:

Petri net with **A**-failures;

A class of Kleene algebras;

GP(**A**)

Conclusions

We achieved

Generalisation of Petri PDL*:

Petri net with **A**-failures;

A class of Kleene algebras;

GP(**A**)

. . . and we may enrich with

Developing a proof calculi and model checking;

Vary the parameter **A** to handle costs and time.

Thank you for your attention!