Réification des accélérations pour la construction de Karp et Miller

Alain Finkel[1], Serge Haddad[1,2], Igor Khmelnitsky[1,2]

(1) LSV ENS Paris-Saclay & CNRS

(2) Inria

Modélisation des Systèmes Réactifs, Angers, 13/11/2019

Plan

1. L'algorithme de Karp et Miller
2. Abstraction et accélération
3. Une preuve simple de l'algorithme de Karp et Miller

Plan

1. L'algorithme de Karp et Miller
2. Abstraction et accélération
3. Une preuve simple de l'algorithme de Karp et Miller

Accessibilité et couverture dans un RdP

- **Accessibilité**

$$Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists \sigma \in T^* \ \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}\}$$

Accessibilité et couverture dans un RdP

- **Accessibilité**

$$Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists\sigma \in T^* \ \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}\}$$

- **Couverture**

$$Cover(\mathcal{N}, \boldsymbol{m}_0) = {\downarrow}Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists\sigma \in T^* \ \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}' \geq \boldsymbol{m}\}$$

Accessibilité et couverture dans un RdP

- **Accessibilité**

$$Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists \sigma \in T^* \ \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}\}$$

- **Couverture**

$$Cover(\mathcal{N}, \boldsymbol{m}_0) = \downarrow Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists \sigma \in T^* \ \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}' \geq \boldsymbol{m}\}$$

- **Représentation finie de la couverture**
  - $\mathbb{N}_\omega = \mathbb{N} \cup \{\omega\}$ et $\mathbb{Z}_\omega = \mathbb{Z} \cup \{\omega\}$
    avec pour tout $n \in \mathbb{Z}$, $\omega > n$ et tout $n \in \mathbb{Z}_\omega$, $\omega + n = \omega$
  - Un ω-*marquage* est un élément de $\mathbb{N}_\omega^P$.
  - Soit $\boldsymbol{m}$ un ω-*marquage*. Alors :

$$[\![\boldsymbol{m}]\!] = \{\boldsymbol{m}' \in \mathbb{N}^P \mid \boldsymbol{m}' \leq \boldsymbol{m}\}$$

Accessibilité et couverture dans un RdP

- **Accessibilité**

$$Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists \sigma \in T^* \; \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}\}$$

- **Couverture**

$$Cover(\mathcal{N}, \boldsymbol{m}_0) = \downarrow Reach(\mathcal{N}, \boldsymbol{m}_0) = \{\boldsymbol{m} \mid \exists \sigma \in T^* \; \boldsymbol{m}_0 \xrightarrow{\sigma} \boldsymbol{m}' \geq \boldsymbol{m}\}$$

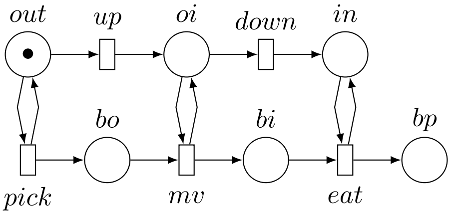
- **Représentation finie de la couverture**
  - $\mathbb{N}_\omega = \mathbb{N} \cup \{\omega\}$ et $\mathbb{Z}_\omega = \mathbb{Z} \cup \{\omega\}$
    avec pour tout $n \in \mathbb{Z}$, $\omega > n$ et tout $n \in \mathbb{Z}_\omega$, $\omega + n = \omega$
  - Un ω-*marquage* est un élément de $\mathbb{N}_\omega^P$.
  - Soit $\boldsymbol{m}$ un ω-*marquage*. Alors :

$$[\![\boldsymbol{m}]\!] = \{\boldsymbol{m}' \in \mathbb{N}^P \mid \boldsymbol{m}' \leq \boldsymbol{m}\}$$

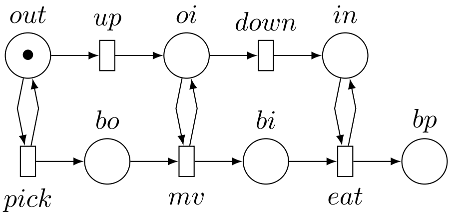
Il existe un ensemble fini (minimal) d'ω-marquages $Clover(\mathcal{N}, \boldsymbol{m}_0)$ tel que :

$$Cover(\mathcal{N}, \boldsymbol{m}_0) = \bigcup_{\boldsymbol{m} \in Clover(\mathcal{N}, \boldsymbol{m}_0)} [\![\boldsymbol{m}]\!]$$

Arbre d'accessibilité d'un RdP

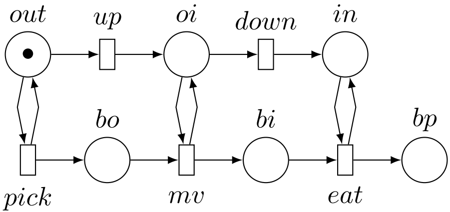


Arbre d'accessibilité d'un RdP



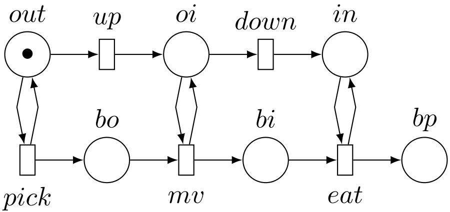
$$out \xrightarrow{up} oi \xrightarrow{down} in$$

Arbre d'accessibilité d'un RdP



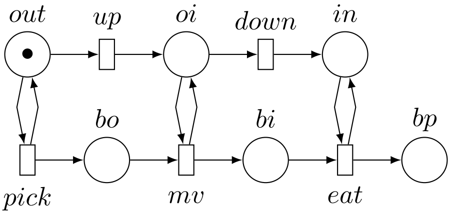
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + bo \arrow[d, "pick"'] & & \\
\bullet\ \bullet\ \bullet & &
\end{tikzcd}
$$

Arbre d'accessibilité d'un RdP



$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + bo \arrow[r, "up"] \arrow[d, "pick"'] & oi + bo \arrow[r, "down"] & in + bo \\
\cdots & &
\end{tikzcd}
$$

Arbre d'accessibilité d'un RdP



$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in & \\
out + bo \arrow[r, "up"] \arrow[d, "pick"'] & oi + bo \arrow[r, "down"] \arrow[d, "mv"'] & in + bo & \\
\bullet \ \bullet \ \bullet & oi + bi \arrow[r, "down"] & in + bi \arrow[r, "eat"] & in + bp
\end{tikzcd}
$$

Construction de l'arbre d'accessibilité

- **Un semi-algorithme** pour construire (V, E, λ, δ)

$u \leftarrow CreerV()$; $\lambda(u) \leftarrow \boldsymbol{m}_0$; $Front \leftarrow \emptyset$; $Inserer(Front, u)$
$V \leftarrow \{u\}$; $E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
 $u \leftarrow ExtraireEquitablement(Front)$
 Pour tout $\lambda(u) \xrightarrow{t} \boldsymbol{m}$ **faire**
  $v \leftarrow CreerV()$; $\lambda(v) \leftarrow \boldsymbol{m}$; $Inserer(Front, v)$
  $V \leftarrow V \cup \{v\}$; $E \leftarrow E \cup \{(u, v)\}$; $\delta(u, v) = t$

Construction de l'arbre d'accessibilité

- **Un semi-algorithme** pour construire (V, E, λ, δ)

$u \leftarrow CreerV()\,;\, \lambda(u) \leftarrow \boldsymbol{m}_0\,;\, Front \leftarrow \emptyset\,;\, Inserer(Front, u)$
$V \leftarrow \{u\}\,;\, E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
  $u \leftarrow ExtraireEquitablement(Front)$
  Pour tout $\lambda(u) \xrightarrow{t} \boldsymbol{m}$ **faire**
    $v \leftarrow CreerV()\,;\, \lambda(v) \leftarrow \boldsymbol{m}\,;\, Inserer(Front, v)$
    $V \leftarrow V \cup \{v\}\,;\, E \leftarrow E \cup \{(u, v)\}\,;\, \delta(u, v) = t$

- **Consistance** $\lambda(r) = \boldsymbol{m}_0$ et pour tout arc $u \xrightarrow{t} v$, on a $\lambda(u) \xrightarrow{t} \lambda(v)$.

Construction de l'arbre d'accessibilité

- **Un semi-algorithme** pour construire (V, E, λ, δ)

$$
\boxed{\begin{array}{l}
u \leftarrow CreerV()\,;\ \lambda(u) \leftarrow \boldsymbol{m}_0\,;\ Front \leftarrow \emptyset\,;\ Inserer(Front, u)\\
V \leftarrow \{u\}\,;\ E \leftarrow \emptyset\\
\textbf{Tant que } Front \neq \emptyset \textbf{ faire}\\
\quad u \leftarrow ExtraireEquitablement(Front)\\
\quad \textbf{Pour tout } \lambda(u) \xrightarrow{t} \boldsymbol{m} \textbf{ faire}\\
\quad\quad v \leftarrow CreerV()\,;\ \lambda(v) \leftarrow \boldsymbol{m}\,;\ Inserer(Front, v)\\
\quad\quad V \leftarrow V \cup \{v\}\,;\ E \leftarrow E \cup \{(u,v)\}\,;\ \delta(u,v) = t
\end{array}}
$$

- **Consistance** $\lambda(r) = \boldsymbol{m}_0$ et pour tout arc $u \xrightarrow{t} v$, on a $\lambda(u) \xrightarrow{t} \lambda(v)$.
- **Complétude** Pour tout $\boldsymbol{m} \in Reach(\mathcal{N}, \boldsymbol{m}_0)$, il existe $u \in V$ tel que :
  - Soit $u \notin Front$ et $\lambda(u) = \boldsymbol{m}$;
  - Soit $u \in Front$ et il existe $\sigma \in T^*$ telle que : $\lambda(u) \xrightarrow{\sigma} \boldsymbol{m}$.

Construction de l'arbre d'accessibilité

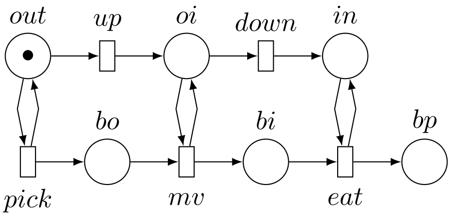
- **Un semi-algorithme** pour construire (V, E, λ, δ)

$u \leftarrow CreerV()\,;\ \lambda(u) \leftarrow \boldsymbol{m}_0\,;\ Front \leftarrow \emptyset\,;\ Inserer(Front, u)$
$V \leftarrow \{u\}\,;\ E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
  $u \leftarrow ExtraireEquitablement(Front)$
  Pour tout $\lambda(u) \xrightarrow{t} \boldsymbol{m}$ **faire**
    $v \leftarrow CreerV()\,;\ \lambda(v) \leftarrow \boldsymbol{m}\,;\ Inserer(Front, v)$
    $V \leftarrow V \cup \{v\}\,;\ E \leftarrow E \cup \{(u, v)\}\,;\ \delta(u, v) = t$

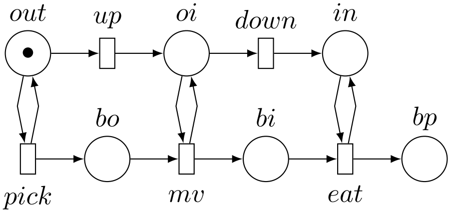
- **Consistance** $\lambda(r) = \boldsymbol{m}_0$ et pour tout arc $u \xrightarrow{t} v$, on a $\lambda(u) \xrightarrow{t} \lambda(v)$.
- **Complétude** Pour tout $\boldsymbol{m} \in Reach(\mathcal{N}, \boldsymbol{m}_0)$, il existe $u \in V$ tel que :
  - Soit $u \notin Front$ et $\lambda(u) = \boldsymbol{m}$;
  - Soit $u \in Front$ et il existe $\sigma \in T^*$ telle que : $\lambda(u) \xrightarrow{\sigma} \boldsymbol{m}$.

Puis on applique l'équité.

Arbre de couverture d'un RdP

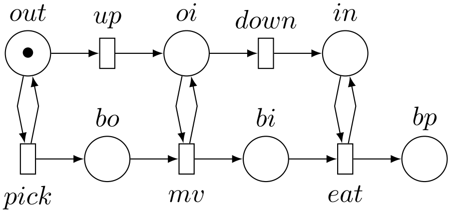


Arbre de couverture d'un RdP



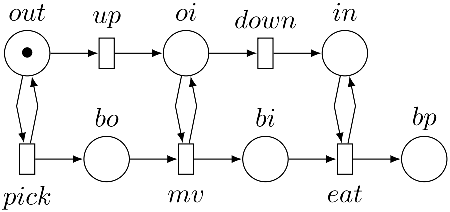
$$out \xrightarrow{\quad up \quad} oi \xrightarrow{\quad down \quad} in$$

Arbre de couverture d'un RdP



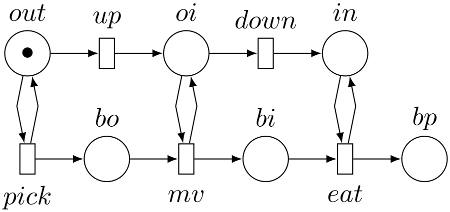
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + bo & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



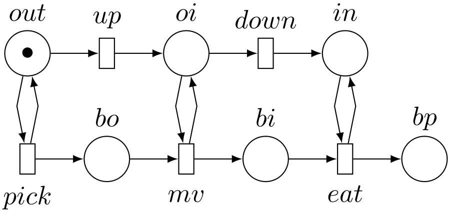
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



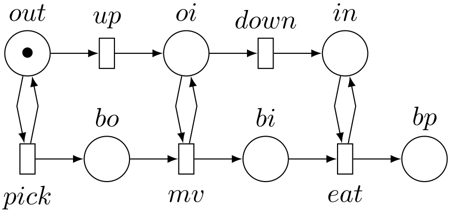
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[d, dotted, "pick"'] & & \\
out + \omega bo & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



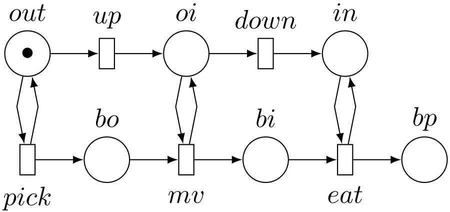
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] & in + \omega bo \\
out + \omega bo & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



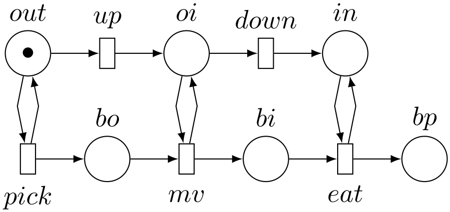
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo \\
out + \omega bo & oi + \omega bo + bi &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



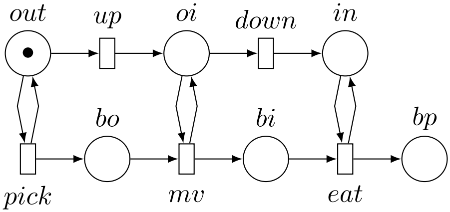
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in+\omega bo \\
out+\omega bo & oi+\omega bo+\omega bi &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



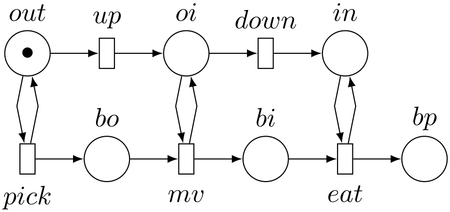
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo \\
out + \omega bo & oi + \omega bo + \omega bi \arrow[d, dotted, "mv"'] & \\
 & oi + \omega bo + \omega bi &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



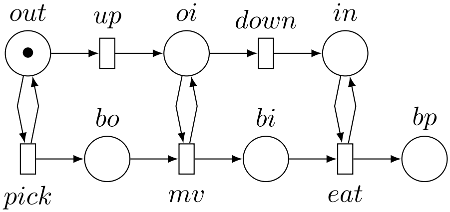
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo \\
out + \omega bo & oi + \omega bo + \omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in + \omega bo + \omega bi \\
 & oi + \omega bo + \omega bi &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



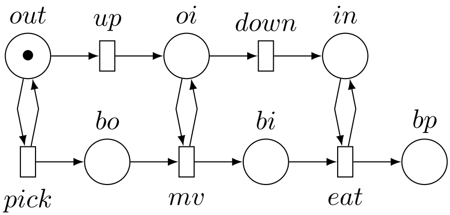
$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in & \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo & \\
out + \omega bo & oi + \omega bo + \omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in + \omega bo + \omega bi \arrow[r, "eat"] & in + \omega bo + \omega bi + bp \\
& oi + \omega bo + \omega bi & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in & \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo & \\
out + \omega bo & oi + \omega bo + \omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in + \omega bo + \omega bi \arrow[r, "eat"] & in + \omega bo + \omega bi + \omega bp \\
 & oi + \omega bo + \omega bi & &
\end{tikzcd}
$$

Arbre de couverture d'un RdP



$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in & \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in+\omega bo & \\
out+\omega bo & oi+\omega bo+\omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in+\omega bo+\omega bi \arrow[r, "eat"] & in+\omega bo+\omega bi+\omega bp \arrow[d, dotted, "eat"'] \\
 & oi+\omega bo+\omega bi & & in+\omega bo+\omega bi+\omega bp
\end{tikzcd}
$$

Construction de l'arbre de couverture

$u \leftarrow CreerV()\,;\ \lambda(u) \leftarrow \boldsymbol{m}_0\,;\ Front \leftarrow \emptyset\,;\ Inserer(Front, u)$
$V \leftarrow \{u\}\,;\ E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
  $u \leftarrow Extraire(Front)$

Construction de l'arbre de couverture

$u \leftarrow CreerV()$; $\lambda(u) \leftarrow \boldsymbol{m}_0$; $Front \leftarrow \emptyset$; $Inserer(Front, u)$
$V \leftarrow \{u\}$; $E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
$u \leftarrow Extraire(Front)$

(1) u est couvert par un ancêtre
Si $\exists v$ ancêtre de u avec $\lambda(v) \geq \lambda(u)$ **alors** $V \leftarrow V \setminus \{u\}$; $E \leftarrow E \setminus V \times \{u\}$

Construction de l'arbre de couverture

$u \leftarrow CreerV()$; $\lambda(u) \leftarrow \boldsymbol{m}_0$; $Front \leftarrow \emptyset$; $Inserer(Front, u)$
$V \leftarrow \{u\}$; $E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
$u \leftarrow Extraire(Front)$

(1) u est couvert par un ancêtre
Si $\exists v$ ancêtre de u avec $\lambda(v) \geq \lambda(u)$ **alors** $V \leftarrow V \setminus \{u\}$; $E \leftarrow E \setminus V \times \{u\}$

(2) $\lambda(u)$ est « accéléré »
Sinon si $\exists v$ ancêtre de u avec $\lambda(v) < \lambda(u)$
et $\exists p$ $\lambda(v)(p) < \lambda(u)(p) < \omega$ **alors**
Pour tout p tel que $\lambda(v)(p) < \lambda(u)(p) < \omega$ **faire** $\lambda(u)(p) \leftarrow \omega$
$Inserer(Front, u)$

Construction de l'arbre de couverture

$u \leftarrow CreerV()$; $\lambda(u) \leftarrow \boldsymbol{m}_0$; $Front \leftarrow \emptyset$; $Inserer(Front, u)$
$V \leftarrow \{u\}$; $E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
$u \leftarrow Extraire(Front)$

(1) u est couvert par un ancêtre
Si $\exists v$ ancêtre de u avec $\lambda(v) \geq \lambda(u)$ **alors** $V \leftarrow V \setminus \{u\}$; $E \leftarrow E \setminus V \times \{u\}$

(2) $\lambda(u)$ est « accéléré »
Sinon si $\exists v$ ancêtre de u avec $\lambda(v) < \lambda(u)$
et $\exists p$ $\lambda(v)(p) < \lambda(u)(p) < \omega$ **alors**
Pour tout p tel que $\lambda(v)(p) < \lambda(u)(p) < \omega$ **faire** $\lambda(u)(p) \leftarrow \omega$
$Inserer(Front, u)$

(3) on explore à partir de u
Sinon
Pour tout $\lambda(u) \xrightarrow{t} \boldsymbol{m}$ **faire**
$v \leftarrow CreerV()$; $\lambda(v) \leftarrow \boldsymbol{m}$; $Inserer(Front, v)$
$V \leftarrow V \cup \{v\}$; $E \leftarrow E \cup \{u \xrightarrow{t} v\}$

Construction de l'arbre de couverture

$u \leftarrow CreerV()$; $\lambda(u) \leftarrow \boldsymbol{m}_0$; $Front \leftarrow \emptyset$; $Inserer(Front, u)$
$V \leftarrow \{u\}$; $E \leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
$u \leftarrow Extraire(Front)$

(1) u est couvert par un ancêtre
Si $\exists v$ ancêtre de u avec $\lambda(v) \geq \lambda(u)$ **alors** $V \leftarrow V \setminus \{u\}$; $E \leftarrow E \setminus V \times \{u\}$

(2) $\lambda(u)$ est « accéléré »
Sinon si $\exists v$ ancêtre de u avec $\lambda(v) < \lambda(u)$
et $\exists p$ $\lambda(v)(p) < \lambda(u)(p) < \omega$ **alors**
Pour tout p tel que $\lambda(v)(p) < \lambda(u)(p) < \omega$ **faire** $\lambda(u)(p) \leftarrow \omega$
$Inserer(Front, u)$

(3) on explore à partir de u
Sinon
Pour tout $\lambda(u) \xrightarrow{t} \boldsymbol{m}$ **faire**
$v \leftarrow CreerV()$; $\lambda(v) \leftarrow \boldsymbol{m}$; $Inserer(Front, v)$
$V \leftarrow V \cup \{v\}$; $E \leftarrow E \cup \{u \xrightarrow{t} v\}$

Consistance ? Pour certains arcs $u \xrightarrow{t} v$, on n'a pas $\lambda(v) \xrightarrow{t} \lambda(w)$.

Plan

ω-transitions

• Syntaxe

Une ω-*transition* $\mathbf{a}$ est définie par :

- son incidence arrière $\mathbf{Pre}(\mathbf{a}) \in \mathbb{N}_\omega^P$;
- son incidence $\mathbf{C}(\mathbf{a}) \in \mathbb{Z}_\omega^P$ avec $\mathbf{Pre}(\mathbf{a}) + \mathbf{C}(\mathbf{a}) \geq 0$ et pour tout p, $\mathbf{Pre}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \mathbf{a}) = \omega$.

ω-transitions

- **Syntaxe**

Une ω-*transition* $\mathbf{a}$ est définie par :

- son incidence arrière $\mathbf{Pre}(\mathbf{a}) \in \mathbb{N}_\omega^P$;
- son incidence $\mathbf{C}(\mathbf{a}) \in \mathbb{Z}_\omega^P$ avec $\mathbf{Pre}(\mathbf{a}) + \mathbf{C}(\mathbf{a}) \geq 0$ et pour tout p, $\mathbf{Pre}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \mathbf{a}) = \omega$.

- **Sémantique**

Soit $\boldsymbol{m} \in \mathbb{N}_\omega^P$. Alors $\mathbf{a}$ est franchissable si $\mathbf{Pre}(\mathbf{a}) \leq \boldsymbol{m}$ et son franchissement conduit à $\boldsymbol{m} + \mathbf{C}(\mathbf{a})$.

ω-transitions

- **Syntaxe**

Une ω-*transition* $\mathbf{a}$ est définie par :

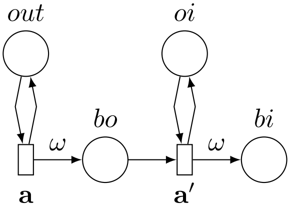
- son incidence arrière $\mathbf{Pre}(\mathbf{a}) \in \mathbb{N}_\omega^P$;
- son incidence $\mathbf{C}(\mathbf{a}) \in \mathbb{Z}_\omega^P$ avec $\mathbf{Pre}(\mathbf{a}) + \mathbf{C}(\mathbf{a}) \geq 0$ et pour tout p, $\mathbf{Pre}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \mathbf{a}) = \omega$.

- **Sémantique**

Soit $\boldsymbol{m} \in \mathbb{N}_\omega^P$. Alors $\mathbf{a}$ est franchissable si $\mathbf{Pre}(\mathbf{a}) \leq \boldsymbol{m}$ et son franchissement conduit à $\boldsymbol{m} + \mathbf{C}(\mathbf{a})$.

- **Concaténation** *(par l'exemple)*

- $\mathbf{Pre}(\mathbf{a}) = out$ et $\mathbf{C}(\mathbf{a}) = \omega bo$
- $\mathbf{Pre}(\mathbf{a}') = oi + bo$ et $\mathbf{C}(\mathbf{a}') = \omega bi - bo$
- $\mathbf{Pre}(\mathbf{a} \cdot \mathbf{a}') = out + oi$ et $\mathbf{C}(\mathbf{a} \cdot \mathbf{a}') = \omega bo + \omega bi$



ω-transitions

- **Syntaxe**

Une ω-*transition* $\mathbf{a}$ est définie par :

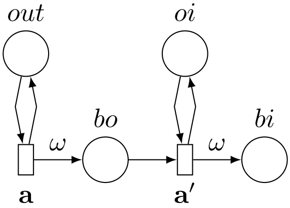
- son incidence arrière $\mathbf{Pre}(\mathbf{a}) \in \mathbb{N}_\omega^P$;
- son incidence $\mathbf{C}(\mathbf{a}) \in \mathbb{Z}_\omega^P$ avec $\mathbf{Pre}(\mathbf{a}) + \mathbf{C}(\mathbf{a}) \geq 0$ et pour tout p, $\mathbf{Pre}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \mathbf{a}) = \omega$.

- **Sémantique**

Soit $\boldsymbol{m} \in \mathbb{N}_\omega^P$. Alors $\mathbf{a}$ est franchissable si $\mathbf{Pre}(\mathbf{a}) \leq \boldsymbol{m}$ et son franchissement conduit à $\boldsymbol{m} + \mathbf{C}(\mathbf{a})$.

- **Concaténation** *(par l'exemple)*

- $\mathbf{Pre}(\mathbf{a}) = out$ et $\mathbf{C}(\mathbf{a}) = \omega bo$
- $\mathbf{Pre}(\mathbf{a}') = oi + bo$ et $\mathbf{C}(\mathbf{a}') = \omega bi - bo$
- $\mathbf{Pre}(\mathbf{a} \cdot \mathbf{a}') = out + oi$ et $\mathbf{C}(\mathbf{a} \cdot \mathbf{a}') = \omega bo + \omega bi$



$$\boldsymbol{m} \xrightarrow{\mathbf{aa}'} \boldsymbol{m}' \text{ si et seulement si } \boldsymbol{m} \xrightarrow{\mathbf{a}\cdot\mathbf{a}'} \boldsymbol{m}'.$$

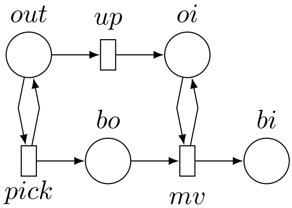
Abstraction de couverture

- Une ω-transition $\mathbf{a}$ est une *abstraction* si pour tout n, il existe $\sigma_n \in T^*$ avec :
  - $\mathbf{Pre}(\sigma_n) \leq \mathbf{Pre}(\mathbf{a})$;
  - Pour tout p, $\mathbf{C}(p, \mathbf{a}) \neq \omega \Rightarrow \mathbf{C}(p, \sigma_n) \geq \mathbf{C}(p, \mathbf{a})$;
  - Pour tout p, $\mathbf{Pre}(p, \mathbf{a}) \neq \omega \wedge \mathbf{C}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \sigma_n) \geq n$.

Abstraction de couverture

- Une ω-transition $\mathbf{a}$ est une *abstraction* si pour tout n, il existe $\sigma_n \in T^*$ avec :
  - $\mathbf{Pre}(\sigma_n) \leq \mathbf{Pre}(\mathbf{a})$;
  - Pour tout p, $\mathbf{C}(p, \mathbf{a}) \neq \omega \Rightarrow \mathbf{C}(p, \sigma_n) \geq \mathbf{C}(p, \mathbf{a})$;
  - Pour tout p, $\mathbf{Pre}(p, \mathbf{a}) \neq \omega \wedge \mathbf{C}(p, \mathbf{a}) = \omega \Rightarrow \mathbf{C}(p, \sigma_n) \geq n$.

- **Illustration**
  - $\mathbf{Pre}(\mathbf{a}) = out$ et $\mathbf{C}(\mathbf{a}) = oi - out + \omega bo + \omega bi$
  - $\sigma_n = pick^{2n}\ up\ mv^n$



Propriétés des abstractions

$\boxed{\text{L'ensemble de couverture est clos par franchissement d'abstraction.}}$

Soit $\mathbf{a}$ une abstraction et $\boldsymbol{m} \in \mathbb{N}_\omega^P$. Si :

- $[\![\boldsymbol{m}]\!] \subseteq Cover(\mathcal{N}, \boldsymbol{m}_0)$
- et $\boldsymbol{m} \xrightarrow{\mathbf{a}} \boldsymbol{m}'$

Alors $[\![\boldsymbol{m}']\!] \subseteq Cover(\mathcal{N}, \boldsymbol{m}_0)$.

Propriétés des abstractions

$\boxed{\text{L'ensemble de couverture est clos par franchissement d'abstraction.}}$

Soit $\mathbf{a}$ une abstraction et $\boldsymbol{m} \in \mathbb{N}_\omega^P$. Si :

- $[\![\boldsymbol{m}]\!] \subseteq Cover(\mathcal{N}, \boldsymbol{m}_0)$
- et $\boldsymbol{m} \xrightarrow{\mathbf{a}} \boldsymbol{m}'$

Alors $[\![\boldsymbol{m}']\!] \subseteq Cover(\mathcal{N}, \boldsymbol{m}_0)$.

$\boxed{\text{L'ensemble des abstractions est clos par concaténation.}}$

Accélération

$\boxed{\mathbf{a} \text{ est une } \textit{accélération} \text{ si } \mathbf{a} \text{ est une abstraction avec } \mathbf{C}(\mathbf{a}) \in \{0, \omega\}^P.}$

Accélération

$\mathbf{a}$ est une *accélération* si $\mathbf{a}$ est une abstraction avec $\mathbf{C}(\mathbf{a}) \in \{0, \omega\}^P$.

- **Comment transformer une abstraction en accélération ?**

Accélération

$\mathbf{a}$ est une *accélération* si $\mathbf{a}$ est une abstraction avec $\mathbf{C}(\mathbf{a}) \in \{0, \omega\}^P$.

• **Comment transformer une abstraction en accélération ?**

Soit $\mathbf{a}$ une abstraction et $\widehat{\mathbf{a}}$ définie par :

- Si $\mathbf{C}(p, \mathbf{a}) < 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{C}(p, \widehat{\mathbf{a}}) = \omega$
- Si $\mathbf{C}(p, \mathbf{a}) = 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{Pre}(p, \mathbf{a})$ et $\mathbf{C}(p, \widehat{\mathbf{a}}) = 0$
- Si $\mathbf{C}(p, \mathbf{a}) > 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{Pre}(p, \mathbf{a})$ et $\mathbf{C}(p, \widehat{\mathbf{a}}) = \omega$

Alors $\widehat{\mathbf{a}}$ est une accélération.

Accélération

$\boxed{\mathbf{a} \text{ est une } \textit{accélération} \text{ si } \mathbf{a} \text{ est une abstraction avec } \mathbf{C}(\mathbf{a}) \in \{0, \omega\}^P.}$

- **Comment transformer une abstraction en accélération ?**

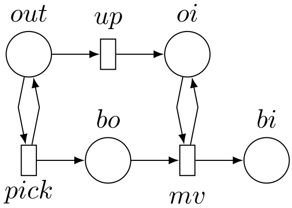
Soit $\mathbf{a}$ une abstraction et $\widehat{\mathbf{a}}$ définie par :

- Si $\mathbf{C}(p, \mathbf{a}) < 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{C}(p, \widehat{\mathbf{a}}) = \omega$
- Si $\mathbf{C}(p, \mathbf{a}) = 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{Pre}(p, \mathbf{a})$ et $\mathbf{C}(p, \widehat{\mathbf{a}}) = 0$
- Si $\mathbf{C}(p, \mathbf{a}) > 0$ alors $\mathbf{Pre}(p, \widehat{\mathbf{a}}) = \mathbf{Pre}(p, \mathbf{a})$ et $\mathbf{C}(p, \widehat{\mathbf{a}}) = \omega$

Alors $\widehat{\mathbf{a}}$ est une accélération.

- **Illustration**

- mv est une transition donc une abstraction.
- $\mathbf{Pre}(\widehat{mv}) = oi + \omega bo$ et $\mathbf{C}(\widehat{mv}) = \omega bo + \omega bi$.



Plan

Un nouvel étiquetage des arcs

Acc (variable fantôme) est un ensemble d'ω-transitions.

. . . Acc $\leftarrow \emptyset$
Tant que $Front \neq \emptyset$ **faire**
$u \leftarrow Extraire(Front)$

Un nouvel étiquetage des arcs

Acc (variable fantôme) est un ensemble d'ω-transitions.

. . . Acc $\leftarrow \emptyset$

Tant que $Front \neq \emptyset$ **faire**

$u \leftarrow Extraire(Front)$

(1) u est couvert par un ancêtre

. . .

(2) $\lambda(u)$ est « accéléré »

Sinon si $\exists v$ ancêtre de u avec $\lambda(v) < \lambda(u)$

et $\exists p$ $\lambda(v)(p) < \lambda(u)(p) < \omega$ **alors**

. . .

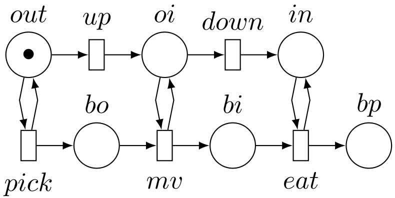
Soit σ la séquence qui étiquette le chemin de v à u et w le père de u

Acc $\leftarrow$ Acc $\cup \{\widehat{\sigma}\}$; $\delta(w,u) \leftarrow \delta(w,u)\widehat{\sigma}$

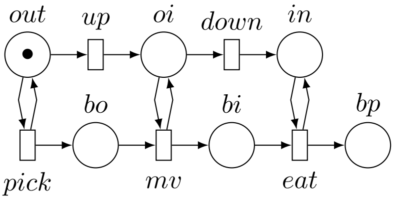
(3) on explore à partir de u

. . .

Arbre de couverture enrichi

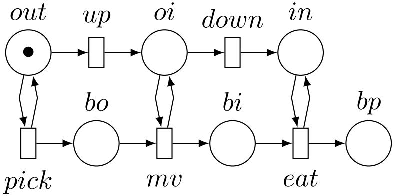


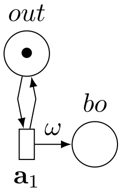
Arbre de couverture enrichi



$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick"'] & oi \arrow[r, "down"] & in \\
out + bo & &
\end{tikzcd}
$$

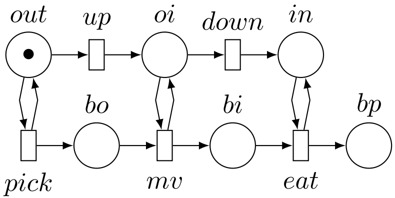
Arbre de couverture enrichi

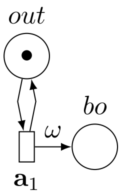




$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out + \omega bo & &
\end{tikzcd}
$$

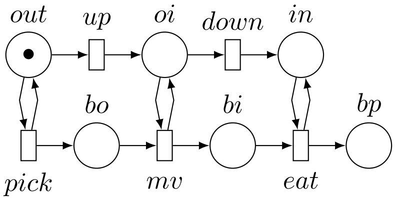
Arbre de couverture enrichi

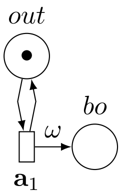




$$\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[d, dotted, "pick"'] & & \\
out + \omega bo & &
\end{tikzcd}$$

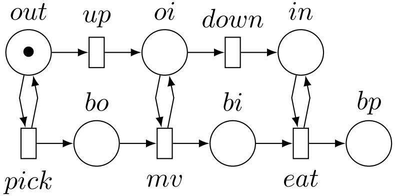
Arbre de couverture enrichi

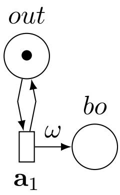




$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, red, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, gray, "pick"'] & oi + \omega bo \arrow[r, "down"] & in + \omega bo \\
\color{gray}{out + \omega bo} & &
\end{tikzcd}
$$

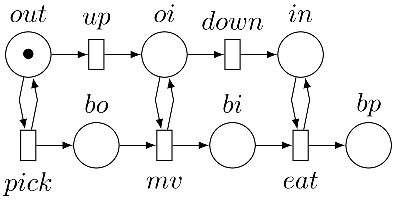
Arbre de couverture enrichi

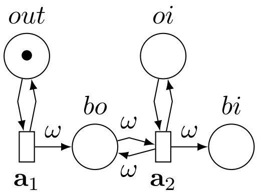




$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, red, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, gray, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv"'] & in + \omega bo \\
\color{gray}{out + \omega bo} & oi + \omega bo + bi &
\end{tikzcd}
$$

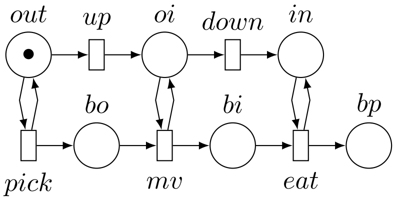
Arbre de couverture enrichi

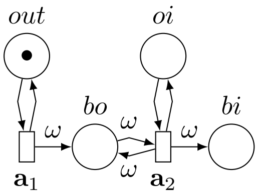




$$\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, "mv\ \mathbf{a}_2"'] & in+\omega bo \\
out+\omega bo & oi+\omega bo+\omega bi &
\end{tikzcd}$$

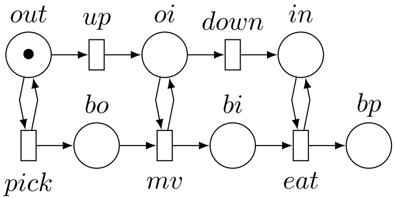
Arbre de couverture enrichi

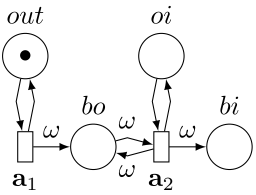




$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out + \omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi + \omega bo \arrow[r, "down"] \arrow[d, "mv\ \mathbf{a}_2"'] & in + \omega bo \\
out + \omega bo & oi + \omega bo + \omega bi \arrow[d, dotted, "mv"'] & \\
 & oi + \omega bo + \omega bi &
\end{tikzcd}
$$

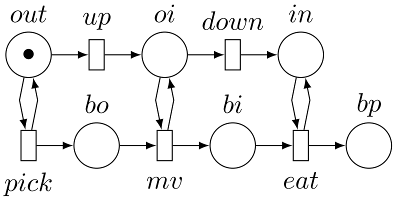
Arbre de couverture enrichi

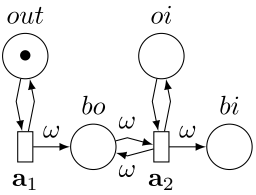




$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, red, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, red, "mv\ \mathbf{a}_2"'] & in+\omega bo \\
out+\omega bo & oi+\omega bo+\omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in+\omega bo+\omega bi \\
 & oi+\omega bo+\omega bi &
\end{tikzcd}
$$

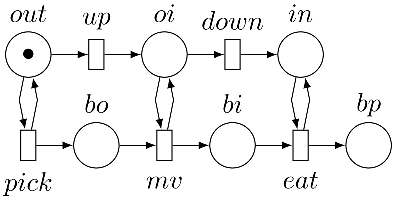
Arbre de couverture enrichi

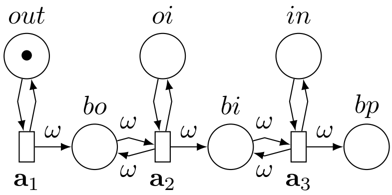




$$\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in & \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, "mv\ \mathbf{a}_2"'] & in+\omega bo & \\
out+\omega bo & oi+\omega bo+\omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in+\omega bo+\omega bi \arrow[r, "eat"] & in+\omega bo+\omega bi+bp \\
 & oi+\omega bo+\omega bi & &
\end{tikzcd}$$

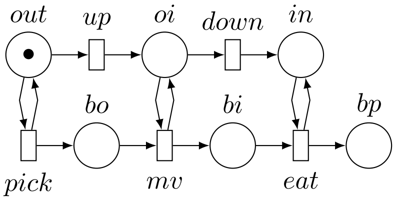
Arbre de couverture enrichi

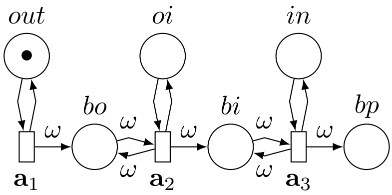




$$\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, red, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in & \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, red, "mv\ \mathbf{a}_2"'] & in+\omega bo & \\
out+\omega bo & oi+\omega bo+\omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in+\omega bo+\omega bi \arrow[r, red, "eat\ \mathbf{a}_3"] & in+\omega bo+\omega bi+\omega bp \\
& oi+\omega bo+\omega bi & &
\end{tikzcd}$$

Arbre de couverture enrichi





$$
\begin{tikzcd}
out \arrow[r, "up"] \arrow[d, red, "pick\ \mathbf{a}_1"'] & oi \arrow[r, "down"] & in & \\
out+\omega bo \arrow[r, "up"] \arrow[d, dotted, "pick"'] & oi+\omega bo \arrow[r, "down"] \arrow[d, red, "mv\ \mathbf{a}_2"'] & in+\omega bo & \\
out+\omega bo & oi+\omega bo+\omega bi \arrow[r, "down"] \arrow[d, dotted, "mv"'] & in+\omega bo+\omega bi \arrow[r, red, "eat\ \mathbf{a}_3"] & in+\omega bo+\omega bi+\omega bp \arrow[d, dotted, "eat"'] \\
 & oi+\omega bo+\omega bi & & in+\omega bo+\omega bi+\omega bp
\end{tikzcd}
$$

Correction de l'algorithme

Terminaison *(par l'absurde)*

- Arbre finiment branchant infini d'où branche infinie ;
- Sous-suite strictement croissante de marquages sur cette branche ... mais chaque marquage a au moins un nouvel ω.

Correction de l'algorithme

Terminaison *(par l'absurde)*

- Arbre finiment branchant infini d'où branche infinie ;
- Sous-suite strictement croissante de marquages sur cette branche ... mais chaque marquage a au moins un nouvel ω.

Consistance

- Pour tout arc (u,v), $\lambda(u) \xrightarrow{\delta(u,v)} \lambda(v)$;
- Acc est un ensemble d'accélérations.

Correction de l'algorithme

Terminaison *(par l'absurde)*

- Arbre finiment branchant infini d'où branche infinie ;
- Sous-suite strictement croissante de marquages sur cette branche ... mais chaque marquage a au moins un nouvel ω.

Consistance

- Pour tout arc (u,v), $\lambda(u) \xrightarrow{\delta(u,v)} \lambda(v)$;
- Acc est un ensemble d'accélérations.

Complétude

$\boldsymbol{m} \xrightarrow{\sigma} \boldsymbol{m}'$ est une *séquence d'exploration* si :

- Il existe $u \in Front$ tel que $\lambda(u) = \boldsymbol{m}$;
- Pour tout $\boldsymbol{m}''$ visité par σ et tout $v \in V \setminus Front$, $\boldsymbol{m}'' \not\leq \lambda(v)$.

Correction de l'algorithme

Terminaison *(par l'absurde)*

- Arbre finiment branchant infini d'où branche infinie ;
- Sous-suite strictement croissante de marquages sur cette branche ... mais chaque marquage a au moins un nouvel ω.

Consistance

- Pour tout arc (u,v), $\lambda(u) \xrightarrow{\delta(u,v)} \lambda(v)$;
- Acc est un ensemble d'accélérations.

Complétude

$\boldsymbol{m} \xrightarrow{\sigma} \boldsymbol{m}'$ est une *séquence d'exploration* si :

- Il existe $u \in Front$ tel que $\lambda(u) = \boldsymbol{m}$;
- Pour tout $\boldsymbol{m}''$ visité par σ et tout $v \in V \setminus Front$, $\boldsymbol{m}'' \not\leq \lambda(v)$.

Pour tout $\boldsymbol{m} \in Cover(\mathcal{N}, \boldsymbol{m}_0)$,

- Soit il existe $u \in V \setminus Front$ tel que $\lambda(u) \geq \boldsymbol{m}$;
- Soit il existe une séquence d'observation $\boldsymbol{m}_1 \xrightarrow{\sigma} \boldsymbol{m}_2 \geq \boldsymbol{m}$.

Conclusion et perspectives

Contributions

- Introduction des abstractions, accélérations et séquences d'exploration ;
- Application à la preuve de la construction de Karp et Miller ;

Conclusion et perspectives

Contributions

- Introduction des abstractions, accélérations et séquences d'exploration ;
- Application à la preuve de la construction de Karp et Miller ;
- Etablissement d'un bel ordre sur les accélérations et borne sur la taille des accélérations minimales ;
- Accélération de la construction de Karp et Miller ...
  à l'aide des accélérations.

Conclusion et perspectives

Contributions

- Introduction des abstractions, accélérations et séquences d'exploration ;
- Application à la preuve de la construction de Karp et Miller ;
- Etablissement d'un bel ordre sur les accélérations et borne sur la taille des accélérations minimales ;
- Accélération de la construction de Karp et Miller ...
  à l'aide des accélérations.

Perspectives

- Application des accélérations pour améliorer les constructions visant à minimiser le nombre maximal d'ω-marquages *(voir la démonstration de l'outil* `MinCov`*)*
- Quid des accélérations dans le cadre plus général des systèmes de transitions bien structurés ?