DM de Langages Formels 2 : Automates pondérés

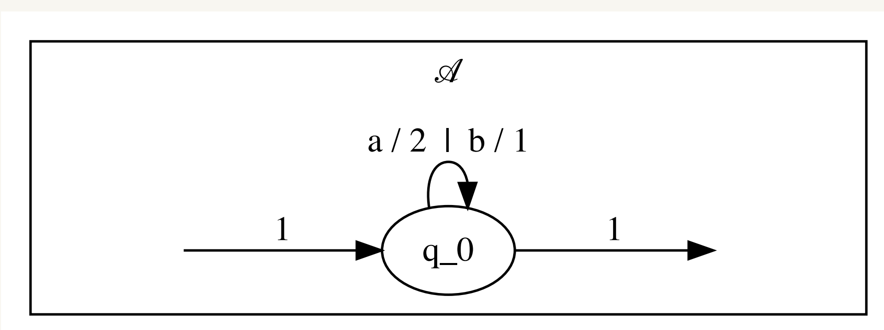
Younesse Kaddar

- [Version PDF]
- http://younesse.net/Logique/DM_LangagesFormels2/

I.

Quelques exemples

1.



En posant :

- $Q \overset{\text{def}}{=} \{q_0\}$
- $\Sigma \overset{\text{def}}{=} \{a, b\}$
- $S \overset{\text{def}}{=} (\mathbb{N}, +, \times, 0, 1)$

$$\lambda(q_0) = \gamma(q_0) \overset{\text{def}}{=} 1$$

$$\mu(q_0, a, q_0) \overset{\text{def}}{=} 2$$

$$\mu(q_0, b, q_0) \overset{\text{def}}{=} 1$$

on vérifie de manière immédiate que :

> $$[\![\mathscr{A}]\!](w) = 2^{|w|_a}$$

2.

Soit $\mathcal{A} \overset{\text{def}}{=} (Q, A, B, q_0, \odot, \circledast, m, \rho)$ un automate séquentiel.

En s'inspirant de l'exemple vu en TD sur le demi-anneau $\langle \mathbb{P}(\Sigma^\ast), \cup, \cdot, \emptyset, \{\varepsilon\} \rangle$, en posant :

- $Q' \overset{\text{def}}{=} Q$
- $\Sigma' \overset{\text{def}}{=} \Sigma$
- $S' \overset{\text{def}}{=} \langle \mathbb{P}(\Sigma^\ast), \cup, \cdot, \emptyset, \{\varepsilon\} \rangle$

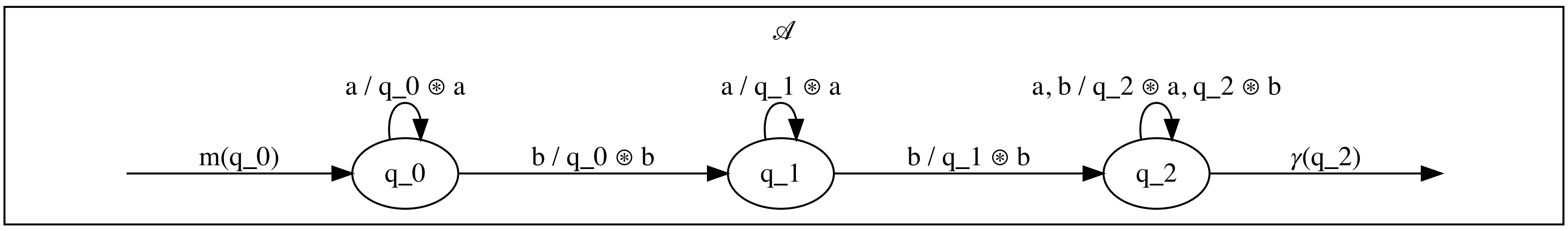
$$\lambda'(q) \overset{\text{def}}{=} \begin{cases} \{m(q)\} \text{ si } m(q) \text{ est défini} \\ \emptyset \text{ sinon} \end{cases}$$

$$\gamma'(q) \overset{\text{def}}{=} \begin{cases} \{\gamma(q)\} \text{ si } \gamma(q) \text{ est défini} \\ \emptyset \text{ sinon} \end{cases}$$

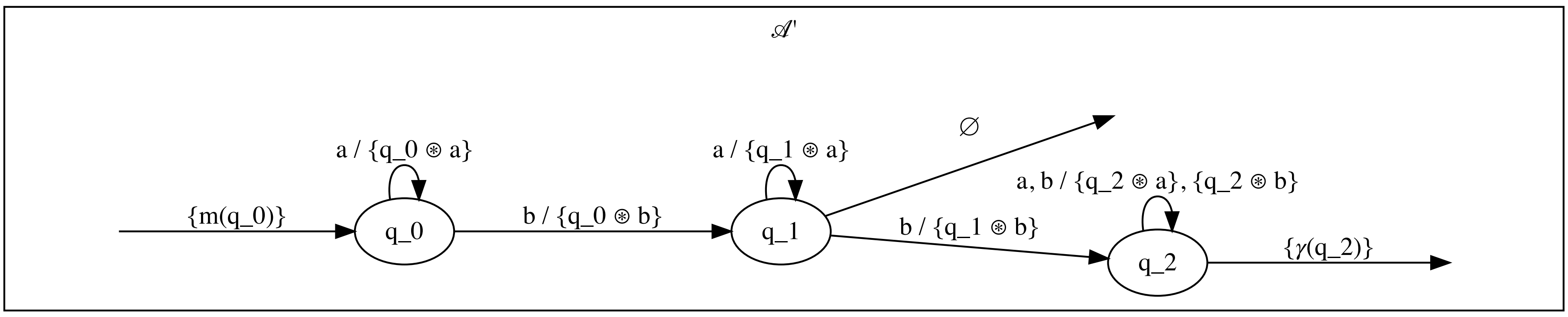
$$\mu(q, a, q') \overset{\text{def}}{=} \begin{cases} \{q \circledast a\} \text{ si } q \overset{a}{\longrightarrow}_{\mathscr{A}} q' \text{ est une transition de } \mathcal{A} \\ \emptyset \text{ sinon} \end{cases}$$

alors

$$[\![\mathcal{A}']\!](w) \overset{\text{def}}{=} \begin{cases} \{[\![\mathcal{A}]\!](w)\} \text{ si } [\![\mathcal{A}]\!](w) \text{ est défini} \\ \emptyset \text{ sinon} \end{cases}$$



$\Downarrow$



Donc avec le demi-anneau S', $[\![\mathcal{A}']\!]$ est une fonction totale qui est en bijection immédiate avec $[\![\mathcal{A}]\!]$ sur le domaine de définition de $[\![\mathcal{A}]\!]$.

Mais on veut l'égalité : on va donc modifier le demi-anneau, tout en restant aussi proche que possible de l'exemple précédent.

On introduit deux nouveaux symboles $\bot$ et $\top$ dénotant respectivement un élément neutre (absorbant pour la deuxième loi) et un élément absorbant pour la première loi, et on pose :

$$S' \overset{\text{def}}{=} \langle \Sigma^* \cup \{\bot, \top\}, +, \cdot, \bot, \varepsilon \rangle$$

où

- $+$ est définie, pour tous $w \neq w' \in \Sigma^*$, de la manière suivante :

$+\ (\nearrow)$	w	w'	$\bot$	$\top$
w	w	$\top$	w	$\top$
w'	$\top$	w'	w'	$\top$
$\bot$	w	w'	$\bot$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$

- $\cdot$ est la concaténation des mots, prolongée par : $\forall w \in \Sigma^*$,
    - $w \cdot \bot = \bot \cdot w = \bot$
    - $w \cdot \top = \top \cdot w = \top$
    - $\bot \cdot \top = \top \cdot \bot = \bot$

On vérifie que S' est bien un demi-anneau :

1. $(\Sigma^* \cup \{\bot, \top\}, +, \bot)$ est bien un monoïde commutatif
    - $+$ est associative : pour tous $x, y, z \in \Sigma^* \cup \{\bot, \top\}$:
        - Si $x = \top$ ou $y = \top$ ou $z = \top$:

        $$x + (y + z) = \top = (x + y) + z$$

        - Sinon si $x = \bot$:

        $$x + (y + z) = y + z = (x + y) + z$$

        - Sinon si $y = \bot$:

        $$x + (y + z) = x + z = (x + y) + z$$

        - Sinon si $z = \bot$:

        $$x + (y + z) = x + y = (x + y) + z$$

        - Sinon si $x = y = z$:

        $$x + (y + z) = x = (x + y) + z$$

        - Sinon :

$$x + (y + z) = \top = (x + y) + z$$

- $+$ est commutative (le tableau est symétrique) et d'élément neutre $\bot$

2. $(\Sigma^*, \cdot, \varepsilon)$ est un monoïde :
   - $\cdot$ est associative : pour tous $x, y, z \in \Sigma^* \cup \{\bot, \top\}$:
     - Si $x = \bot$ ou $y = \bot$ ou $z = \bot$:

$$x \cdot (y \cdot z) = \bot = (x \cdot y) \cdot z$$

     - Sinon si $x = \top$ ou $y = \top$ ou $z = \top$:

$$x \cdot (y \cdot z) = \top = (x \cdot y) \cdot z$$

     - Sinon (associativité de la concaténation pour les mots) :

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z$$

   - $\cdot$ a pour élément neutre ε

3. $\cdot$ est distributive par rapport à $+$: pour tous $x, y, z \in \Sigma^* \cup \{\bot, \top\}$:
   - Si $x = \bot$:

$$\begin{aligned} \bot \cdot (y + z) = \bot = \bot \cdot y + \bot \cdot z \\ (y + z) \cdot \bot = \bot = y \cdot \bot + z \cdot \bot \end{aligned}$$

   - Sinon si $y = z = \bot$:

$$\begin{aligned} x \cdot (y + z) = \bot = x \cdot y + x \cdot z \\ (y + z) \cdot x = \bot = y \cdot x + z \cdot x \end{aligned}$$

   - Sinon si $x = \top$:

$$\begin{aligned} \top \cdot (y + z) = \top = \top \cdot y + \top \cdot z \\ (y + z) \cdot \top = \top = y \cdot \top + z \cdot \top \end{aligned}$$

     car l'un des deux derniers termes (à droite) vaut $\top$
   - Sinon si $y = z$ ou $z = \bot$:

$$\begin{aligned} x \cdot (y + z) = x \cdot y = x \cdot y + x \cdot z \\ (y + z) \cdot x = y \cdot x = y \cdot x + z \cdot x \end{aligned}$$

   - Sinon si $y = \bot$:

$$\begin{aligned} x \cdot (y + z) = x \cdot z = x \cdot y + x \cdot z \\ (y + z) \cdot x = z \cdot x = y \cdot x + z \cdot x \end{aligned}$$

   - Sinon ($x \in \Sigma^*, y, z \in \Sigma^* \cup \{\top\}$ et $y \neq z$) :

$$\begin{aligned} x \cdot (y + z) = x \cdot \top = \top = x \cdot y + x \cdot z \\ (y + z) \cdot x = \top \cdot x = \top = y \cdot x + z \cdot x \end{aligned}$$

4. $\bot$ est absorbant pour $\cdot$

Il s'ensuit qu'en posant :

- $Q' \overset{\text{def}}{=} Q$
- $\Sigma' \overset{\text{def}}{=} \Sigma$
- $S' \overset{\text{def}}{=} \langle \Sigma^* \cup \{\bot, \top\}, +, \cdot, \bot, \varepsilon \rangle$

$$\lambda'(q) \overset{\text{def}}{=} \begin{cases} m(q) \text{ si } m(q) \text{ est défini} \\ \bot \text{ sinon} \end{cases}$$

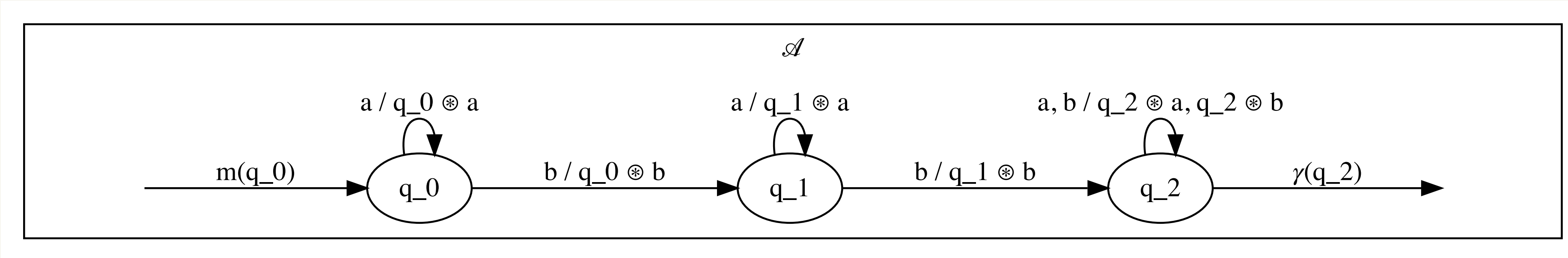
$$\gamma'(q) \overset{\text{def}}{=} \begin{cases} \gamma(q) \text{ si } \gamma(q) \text{ est défini} \\ \bot \text{ sinon} \end{cases}$$

$$\mu(q, a, q') \overset{\text{def}}{=} \begin{cases} (q \circledast a) \text{ si } q \xrightarrow{a}_{\mathscr{A}} q' \text{ est une transition de } \mathcal{A} \\ \bot \text{ sinon} \end{cases}$$

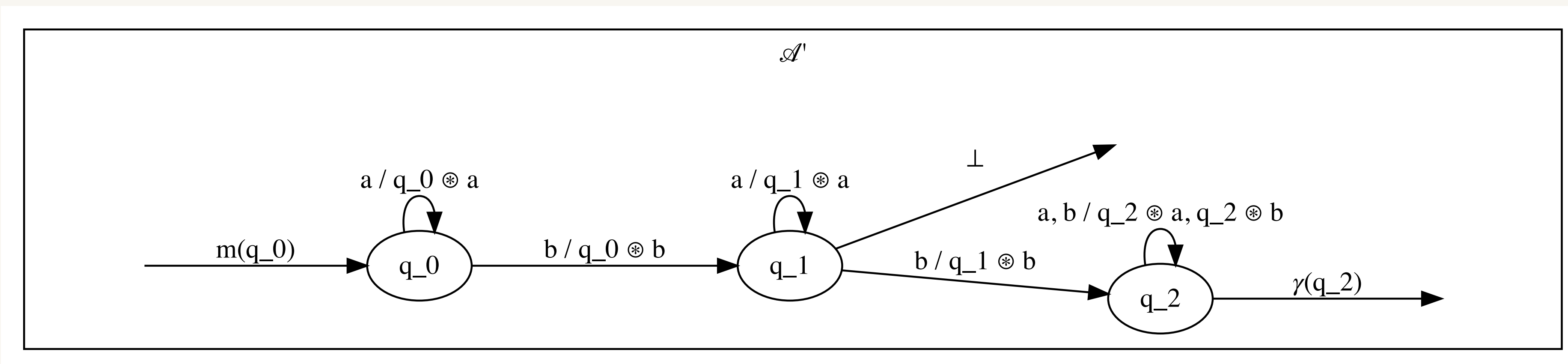
alors

> $$[\![\mathcal{A}']\!](w) \overset{\text{def}}{=} \begin{cases} [\![\mathcal{A}]\!](w) \text{ si } [\![\mathcal{A}]\!](w) \text{ est défini} \\ \bot \text{ sinon} \end{cases}$$

(la démonstration est analogue à ce qu'on a vu en TD : on le vérifie de manière immédiate, puisque $\bot$ est absorbant, et qu'on n'a qu'un seul chemin au plus dans $\mathcal{A}$ acceptant w, par déterminisme de la fonction séquentielle)

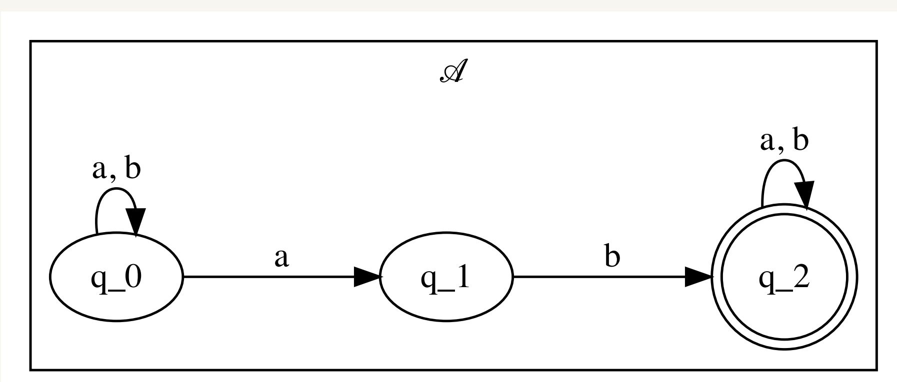


$\Downarrow$

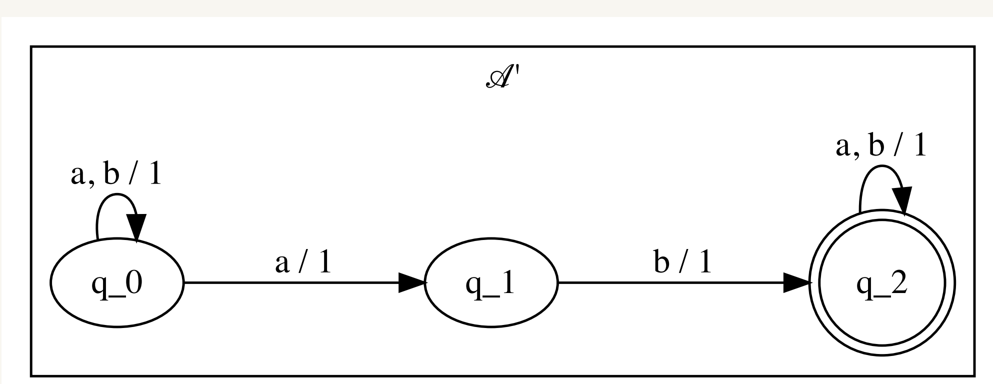


3.

$\mathcal{A} \stackrel{\text{def}}{=} (\Sigma, Q, \delta, I, F)$



$\Downarrow$



En posant :

- $Q' \stackrel{\text{def}}{=} Q$
- $\Sigma' \stackrel{\text{def}}{=} \Sigma$
- $S' \stackrel{\text{def}}{=} \langle \mathbb{N}, +, \times, 0, 1 \rangle$

$$\lambda'(q) \stackrel{\text{def}}{=} \begin{cases} 1 \text{ si } q \text{ est initial} \\ 0 \text{ sinon} \end{cases}$$

$$\gamma'(q) \stackrel{\text{def}}{=} \begin{cases} 1 \text{ si } q \text{ est final} \\ 0 \text{ sinon} \end{cases}$$

$$\mu(q, a, q') \stackrel{\text{def}}{=} \begin{cases} 1 \text{ si } q' \in \delta(q, a) \\ 0 \text{ sinon} \end{cases}$$

alors

$$\begin{aligned} [\![\mathcal{A}']\!](w) &\stackrel{\text{def}}{=} \sum_{\underbrace{q_1}_{\in I} \xrightarrow{w_1}_{\mathcal{A}} q_2 \longrightarrow \cdots \xrightarrow{w_n}_{\mathcal{A}} \underbrace{q_{n+1}}_{\in F}} \underbrace{1 \times \cdots \times 1}_{n+1 \text{ fois}} \\ &= \left| \left\{ C \stackrel{\text{def}}{=} \underbrace{q_1}_{\in I} \xrightarrow{w_1}_{\mathcal{A}} q_2 \longrightarrow \cdots \xrightarrow{w_n}_{\mathcal{A}} \underbrace{q_{n+1}}_{\in F} \mid q_i \in Q \right\} \right| \\ &= \left| \left\{ C \text{ chemin acceptant étiqueté par } w \right\} \right| \\ &= p_{\mathcal{A}}(w) \end{aligned}$$

4.

On note S le demi-anneau produit $\{♀, ☠, ❓, ⦸\}^{|V|}$ (non commutatif) muni des lois :

$\times$ ($\nearrow$)	♀	☠	❓	⦸
♀	♀	♀	♀	⦸
☠	☠	☠	☠	⦸
❓	♀	☠	❓	⦸
⦸	⦸	⦸	⦸	⦸

$+$ ($\nearrow$)	♀	☠	❓	⦸
♀	♀	♀	♀	♀
☠	♀	☠	☠	☠
❓	♀	☠	❓	❓
⦸	♀	☠	❓	⦸

définies **composante par composante**, où :

- ♀ sera interprété comme “la variable est vivante”
- ☠ sera interprété comme “la variable est morte”
- ❓ sera interprété comme “la variable est dans un état indéterminé”
- ⦸ est un élément neutre (pour $+$) absorbant rajouté artificiellement

NB : Par abus de notation, on notera de la même manière les lois définies sur $\{♀, ☠, ❓, ⦸\}$ et les lois produit correspondantes sur $\{♀, ☠, ❓, ⦸\}^{|V|}$

On vérifie bien que

- $(S, +)$ est un monoïde commutatif d’élément neutre ⦸
- $(S, \times)$ est un monoïde d’élément neutre ❓
- $\times$ est distributive par rapport à $+$:
    - il suffit de le vérifier composante par composante : pour tous $x, y, z \in \{♀, ☠, ❓, ⦸\}$:
        - Si $x = ⦸$:

$$
\begin{aligned}
⦸ \times (y+z) = ⦸ = ⦸ + ⦸ = ⦸ \times y + ⦸ \times z \\
(y+z) \times ⦸ = ⦸ = ⦸ + ⦸ = y \times ⦸ + z \times ⦸
\end{aligned}
$$

        - Sinon si $x =$ ❓ :

$$
\begin{aligned}
❓ \times (y+z) = y + z = ❓ \times y + ❓ \times z \\
(y+z) \times ❓ = y + z = y \times ❓ + z \times ❓
\end{aligned}
$$

        - Sinon si $y = z = ⦸$:

$$
\begin{aligned}
x \times (y+z) = ⦸ = x \times y + x \times z \\
(y+z) \times x = ⦸ = y \times x + z \times x
\end{aligned}
$$

        - Sinon :

$$
\begin{aligned}
x \times (y+z) = x = x \times y + x \times z \\
(y+z) \times x = y + z = y \times x + z \times x
\end{aligned}
$$

- ⦸ est absorbant
- S est **borné idempotent** :
    - $+$ est idempotent
    - Comme $⦸ \sqsubseteq$ ❓ $\sqsubseteq ☠ \sqsubseteq ♀$, S ne contient aucune chaîne infinie décroissante non stationnaire.

> Donc S est un demi-anneau borné idempotent.

On pose $G' \overset{\text{def}}{=} (Q, \Sigma, S, \lambda, \mu, \gamma)$, où

- L’ensemble des états Q est l’ensemble des sommets de G
- $\Sigma \overset{\text{def}}{=} I(V)$
- $S \overset{\text{def}}{=} (\{♀, ☠, ❓, ⦸\}^{|V|}, +, \times, ⦸, ❓)$

Dans la suite, on notera $x_1, \ldots, x_n$ les variables de V, et on indicera les vecteurs par i pour dénoter leur coordonnée selon x_i.

Pour tous $e \in I(V), q, q' \in Q, i \in [\![1, n]\!]$:

$$\lambda(q) \overset{\text{def}}{=} (\,\textbf{?}, \ldots, \textbf{?}\,)$$

$$\gamma(q) \overset{\text{def}}{=} \begin{cases} (\text{☠}, \ldots, \text{☠}) \text{ si } q = q_f \\ (\,\textbf{?}, \ldots, \textbf{?}\,) \text{ sinon} \end{cases}$$

$$\mu(q, e, q')_i \overset{\text{def}}{=} \begin{cases} \text{♀} & \begin{array}{l} \text{si } q \longrightarrow^{e} q' \text{ est une arête de } G \\ \text{et } e \text{ est de la forme} \\ \text{if e'(V') ou y := e'(V') avec } y \in V, x_i \in V' \end{array} \\ \\ \text{☠} & \begin{array}{l} \text{sinon si } q \longrightarrow^{e} q' \text{ est une arête de } G \\ \text{et } e \text{ est de la forme} \\ x_i := \text{e'(V') avec } V' \subseteq V \end{array} \\ \\ \textbf{?} \text{ sinon} & \end{cases}$$

> Pour tout chemin
>
> $$C \overset{\text{def}}{=} q_{j_1} \longrightarrow^{e_{j_1}} q_{j_2} \longrightarrow^{e_{j_2}} \cdots \longrightarrow^{e_{j_{m-1}}} q_{j_m} \overset{\text{def}}{=} q_f$$
>
> montrons que la projection sur la i-ème composante de $\|C\|$ permet de déterminer l'état de la variable x_i

- *Cas 1* : *Si* pour tout $k \in [\![1, m-1]\!]$, x_i n'apparaît pas dans e_{j_k} :
  alors x_i est morte dans C, et :

$$\begin{aligned} \|C\| &= \lambda(q_{j_1}) \prod_{k=1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) \gamma(q_f) \\ &= (\,\textbf{?}, \ldots, \textbf{?}\,) \left(\prod_{k=1}^{m-1} (\,\textbf{?}, \ldots, \textbf{?}\,) \right) (\text{☠}, \ldots, \text{☠}) \\ &= (\text{☠}, \ldots, \text{☠}) \end{aligned}$$

  donc on a bien $\|C\|_i = \text{☠}$

- *Cas 2* : *Sinon*, on note l le plus entier de $[\![1, m-1]\!]$ tel que x_i apparaît dans e_{j_l} : autrement dit, e_{j_l} est la première instruction du chemin C où apparaît x_i
  - *Sous-Cas 1* : si e_{j_l} est de la forme if e'(V') ou y := e'(V'), avec $y \in V, x_i \in V'$: *alors* x_i est vivante dans C, et :

$$\begin{aligned} \|C\|_i &= \left[\lambda(q_{j_1}) \prod_{k=1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) \gamma(q_f) \right]_i \\ &= \left[\left((\,\textbf{?}, \ldots, \textbf{?}\,) \prod_{k=1}^{l-1} \underbrace{\mu(q_{j_k}, e_{j_k}, q_{j_{k+1}})}_{=(\,\textbf{?}, \ldots, \textbf{?}\,)} \right) \mu(q_{j_l}, e_{j_l}, q_{j_{l+1}}) \prod_{k=l+1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) (\text{☠}, \ldots, \text{☠}) \right]_i && \text{(associativité)} \\ &= \overbrace{\left[\left((\,\textbf{?}, \ldots, \textbf{?}\,) \prod_{k=1}^{l-1} (\,\textbf{?}, \ldots, \textbf{?}\,) \right) \right]_i}^{=\,\textbf{?}} \quad \underbrace{\left[\mu(q_{j_l}, e_{j_l}, q_{j_{l+1}}) \right]_i}_{=\,\text{♀}} \left[\prod_{k=l+1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) (\text{☠}, \ldots, \text{☠}) \right]_i \\ &= \underbrace{\left[\mu(q_{j_l}, e_{j_l}, q_{j_{l+1}}) \right]_i}_{=\,\text{♀}} \underbrace{\left[\prod_{k=l+1}^{m-1} \underbrace{\mu(q_{j_k}, e_{j_k}, q_{j_{k+1}})}_{\neq \text{⦸}} (\text{☠}, \ldots, \text{☠}) \right]_i}_{\neq \text{⦸}} \\ &= \text{♀} \end{aligned}$$

  - *Sous-Cas 2* : si e_{j_l} est de la forme $x_i :=$ e'(V'), avec $V' \subseteq V$: *alors* x_i est morte dans C, et :

$$\begin{aligned} \|C\|_i &= \left[\lambda(q_{j_1}) \prod_{k=1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) \gamma(q_f) \right]_i \\ &= \overbrace{\left[\left((\,\textbf{?}, \ldots, \textbf{?}\,) \prod_{k=1}^{l-1} (\,\textbf{?}, \ldots, \textbf{?}\,) \right) \right]_i}^{=\,\textbf{?}} \quad \underbrace{\left[\mu(q_{j_l}, e_{j_l}, q_{j_{l+1}}) \right]_i}_{=\,\text{☠}} \left[\prod_{k=l+1}^{m-1} \mu(q_{j_k}, e_{j_k}, q_{j_{k+1}}) (\text{☠}, \ldots, \text{☠}) \right]_i \\ &= \text{☠} \underbrace{\left[\prod_{k=l+1}^{m-1} \underbrace{\mu(q_{j_k}, e_{j_k}, q_{j_{k+1}})}_{\neq \text{⦸}} (\text{☠}, \ldots, \text{☠}) \right]_i}_{\neq \text{⦸}} \\ &= \text{☠} \end{aligned}$$

Dans tous les cas, le résultat est acquis.

> Pour tout sommet $q \in Q$, montrons que la projection sur la i-ème composante de
>
> $$\|q\| \overset{\text{def}}{=} \sum_{C \text{ chemin partant de } q} \|C\|$$
>
> permet de déterminer l'état de la variable x_i

Supposons qu'il existe un chemin C de q vers q_f où x_i est vivante.

Alors par commutativité de $+$:

$$\underbrace{\|q\|_i = \underbrace{\|C\|_i}_{= ♀} + \sum_{C' \neq C \text{ chemin partant de } q} \|C'\|_i}_{= ♀}$$

et le résultat est acquis.

Réciproquement, s'il n'existe aucun chemin de q vers q_f où x_i est vivante : alors on ne peut pas avoir $\|q\|_i = ♀$ puisque $+(\{☠, ❓, ⦰\}^2) \subseteq \{☠, ❓, ⦰\}$, donc une somme d'éléments différents de ♀ est différente de ♀.

> Il vient donc que, pour tout sommet q et chemin C, les variables vivantes de $\|q\|$ et de $\|C\|$ sont celles dont la coordonnée vaut ♀.

5.

En se basant pour les observations faites précédemment, il apparaît que pour toute variable x_i et pour tout état q :

1. x_i **est vivante dans** q si et seulement si il existe un chemin

   $$q \overset{\text{def}}{=} q_{j_1} \longrightarrow^{e_{j_1}} q_{j_2} \longrightarrow^{e_{j_2}} \cdots \longrightarrow^{e_{j_{m-1}}} q_{j_m} \overset{\text{def}}{=} q_f$$

   tel qu'en notant l le plus entier de $[\![1, m-1]\!]$ tel que x_i apparaît dans e_{j_l} (*i.e.* e_{j_l} est la *première* instruction du chemin où apparaît x_i), e_{j_l} est de la forme

   $$\text{if e'(V')} \text{ ou } \text{y := e'(V')}$$

   avec $y \in V, x_i \in V'$

2. x_i **est morte dans** q si et seulement si elle n'est pas vivante dans q et
   - il existe un chemin

     $$q \overset{\text{def}}{=} q_{j_1} \longrightarrow^{e_{j_1}} q_{j_2} \longrightarrow^{e_{j_2}} \cdots \longrightarrow^{e_{j_{m-1}}} q_{j_m} \overset{\text{def}}{=} q_f$$

     tel que la *première* instruction du chemin où apparaît x_i est de la forme

     $$x_i := \text{e'(V')}$$

     avec $V' \subseteq V$

   ou

   - x_i n'apparaît dans aucun chemin de q à q_f

Si on veut que $\|q\|_i$ indique l'état de la variable x_i dans l'état q, chaque transition $q \longrightarrow^e_G q'$ doit donc vérifier les propriétés suivantes :

> 1. $P_{♀}\left(q \longrightarrow^e_G q', i\right)$:
>   si $\|q\|_i \neq ♀$ et e est de la forme
>
> $$\text{if e'(V')} \text{ ou } \text{y := e'(V')}$$
>
>   avec $y \in V, x_i \in V'$, alors
>
> $$\|q\|_i = ♀$$
>
> 2. $P_{☠}\left(q \longrightarrow^e_G q', i\right)$:
>   si $\|q\|_i \neq ♀$ et e est de la forme
>
> $$x_i := \text{e'(V')}$$
>
>   avec $V' \subseteq V$, alors
>
> $$\|q\|_i = ☠$$
>
> 3. $P_{❓}\left(q \longrightarrow^e_G q', i\right)$:
>   si $\|q\|_i \neq ♀$ ou $\|q'\|_i \neq ♀$ et x_i n'apparaît pas dans e, alors
>
> $$\|q\|_i = \|q'\|_i$$

On vérifie aussi que ces conditions sont suffisantes, d'après les observations précédentes.

De fait, la correction de l'algorithme suivant s'ensuit :

```
Initialiser tous les ||q||_i à ☠

Pour chaque x_i:
  Tant qu'il existe un transition t ≝ (q1 ⟶^e q') qui ne vérifie pas P_❓(t,i) ou P_☥(t,i):
      mettre ||q||_i ou ||q'||_i à ☥ pour vérifier la condition
```

NB : comme, à chaque étape de l'algorithme, les $\Vert q\Vert_i$ valent soit ☠, soit ♀ : les conditions $P_{\text{☠}}\left(q \overset{e}{\longrightarrow}_G q', i\right)$ sont toujours vérifiées, donc il est inutile d'en faire mention dans la boucle interne.

Terminaison

L'algorithme termine clairement, car

- toute transition $t \overset{\text{def}}{=} q \overset{e}{\longrightarrow}_G q'$ telle que $\Vert q\Vert_i = \text{♀}$ vérifie $P_{\text{♀}}(t, i)$
- toute transition $t \overset{\text{def}}{=} q \overset{e}{\longrightarrow}_G q'$ telle que $\Vert q\Vert_i = \Vert q\Vert_i \in \lbrace \text{♀}, \text{☠}\rbrace$ vérifie $P_{\text{❓}}(t, i)$

et chaque $\Vert q\Vert_i$ ne peut être modifié (pour se voir attribuer la valeur ♀) qu'au plus une fois dans la boucle principale.

Complexité

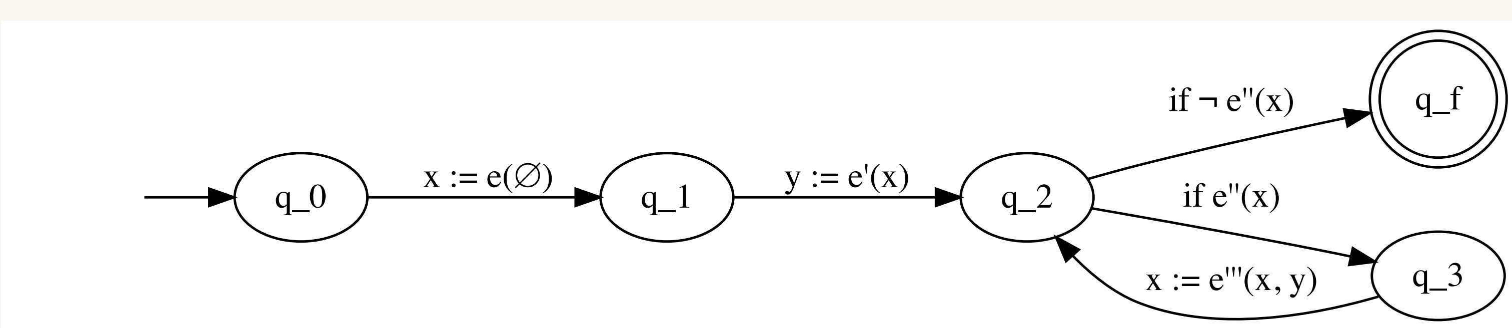
Dans le pire des cas, il faudra modifier chacun des $\Vert q\Vert_i$, pour tous q, i : et à chaque modification, on cherche une transition qui transgresse une des trois propriétés (en temps $O(\vert\delta\vert m)$ où δ est l'ensemble des transitions et m la taille maximale des instructions $e \in I(V)$) donc

> la complexité de l'algorithme est en
>
> $$O(\vert V\vert \vert Q\vert \vert\delta\vert m)$$
>
> où
>
> - Q est l'ensemble des états
> - δ est l'ensemble des transitions
> - m est la taille maximale d'une instruction de $I(V)$

NB : en fait, on ne modifiera jamais les $\Vert q_f\Vert_i$ puisqu'il n'y a pas de transition sortante de q_f, mais on ne cherche pas à être aussi précis.

Exemple :

```
x := 0
y := x+1
if x == 0:
  x := x+y
  goto line 2
```

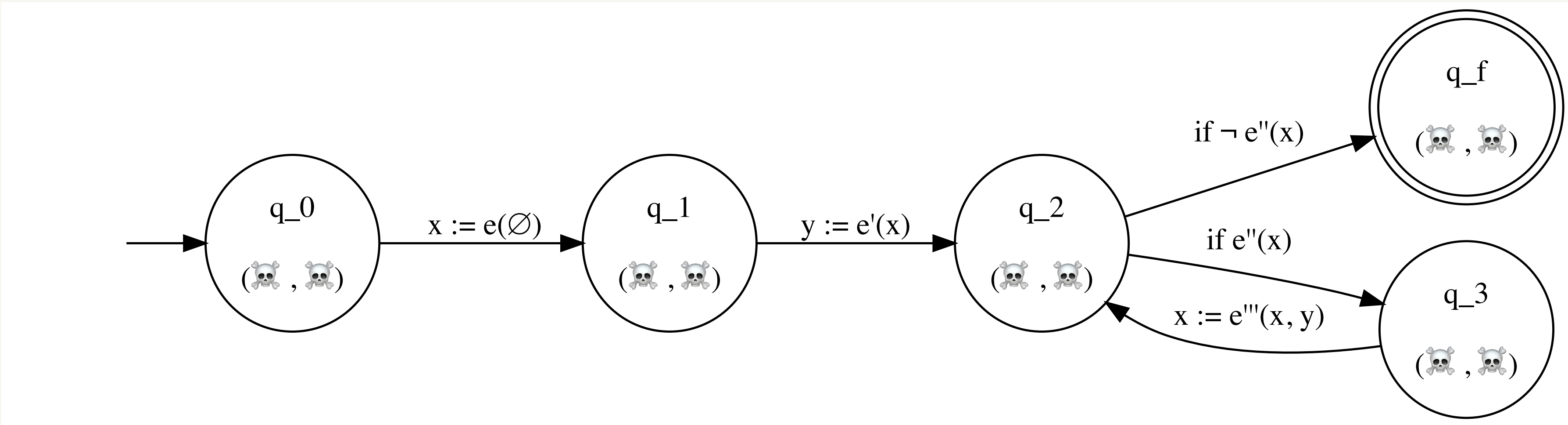


$$I(V) = \Big\lbrace \text{x := e}(\emptyset),\ \text{y := e'(x)},\ \text{if e''(x)},\ \text{if} \neg \text{e''(x)},\ \text{x := e'''(x, y)}\Big\rbrace$$

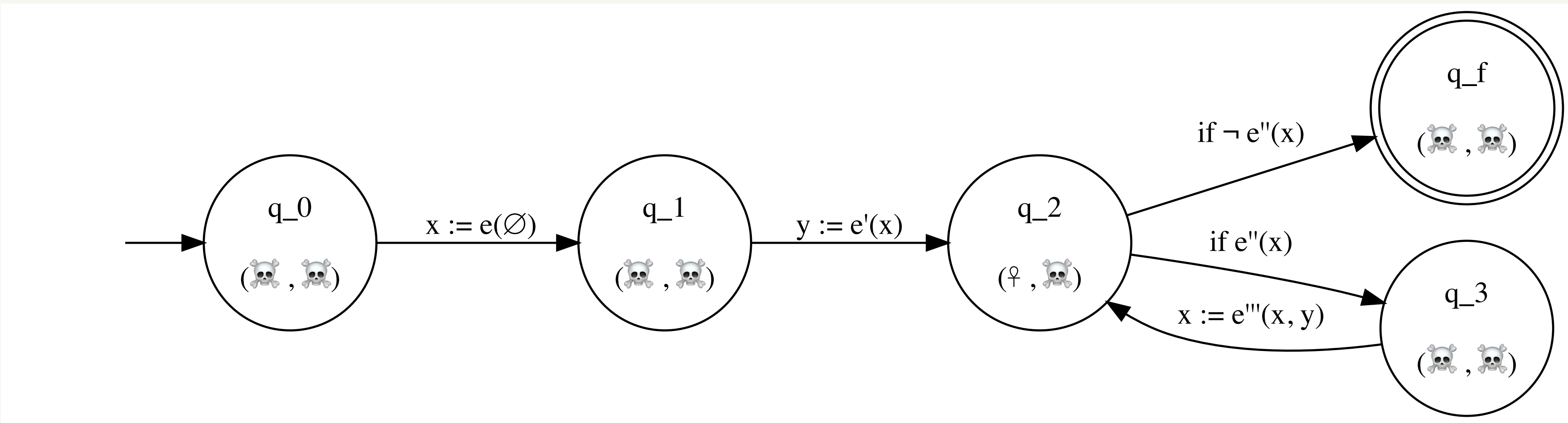
On pose $x_1 \overset{\text{def}}{=} x, x_2 \overset{\text{def}}{=} y$

- *Initialisation* :

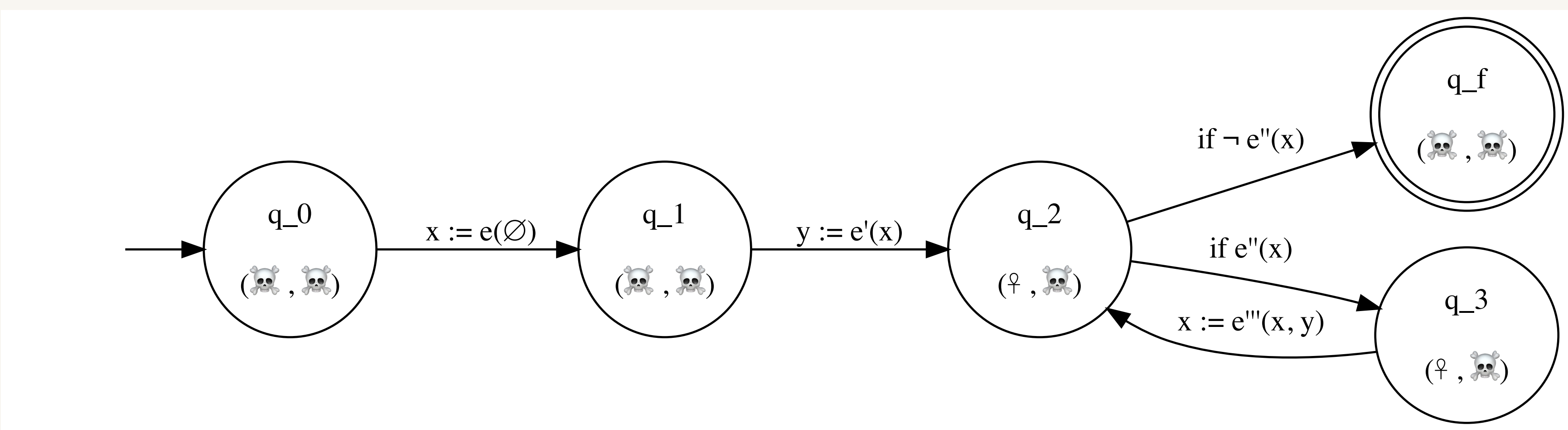
q	$\Vert q\Vert$
q_0	(☠, ☠)
q_1	(☠, ☠)
q_2	(☠, ☠)
q_3	(☠, ☠)
q_f	(☠, ☠)



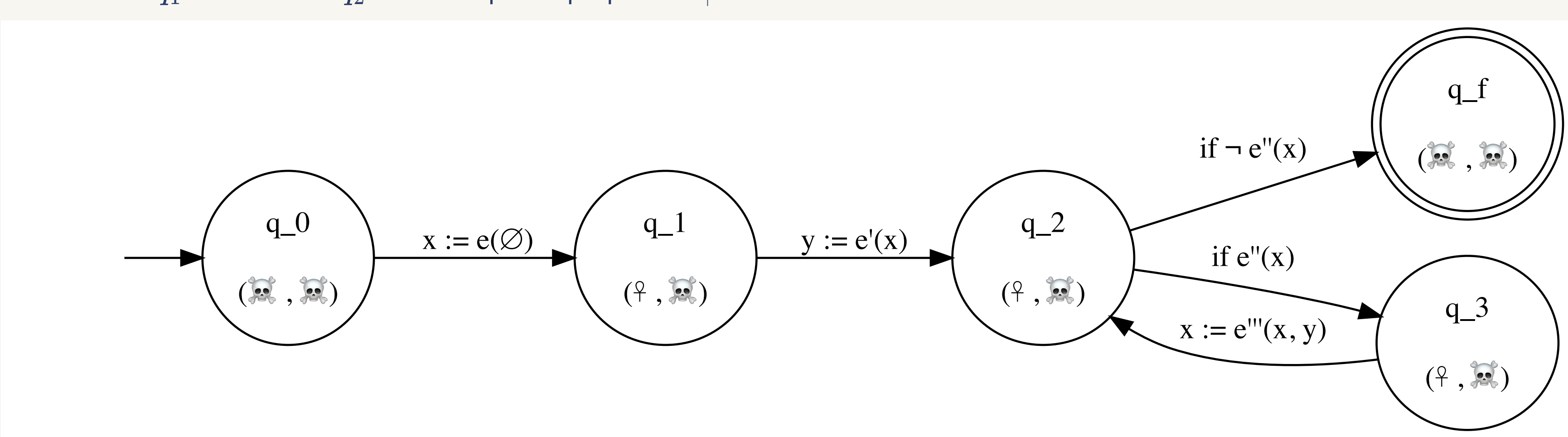
- $x_i = x$:
  - La transition $q_2 \longrightarrow^{\text{if} \neg e''(x)} q_f$ ne vérifie pas la propriété $P_{♀}$:



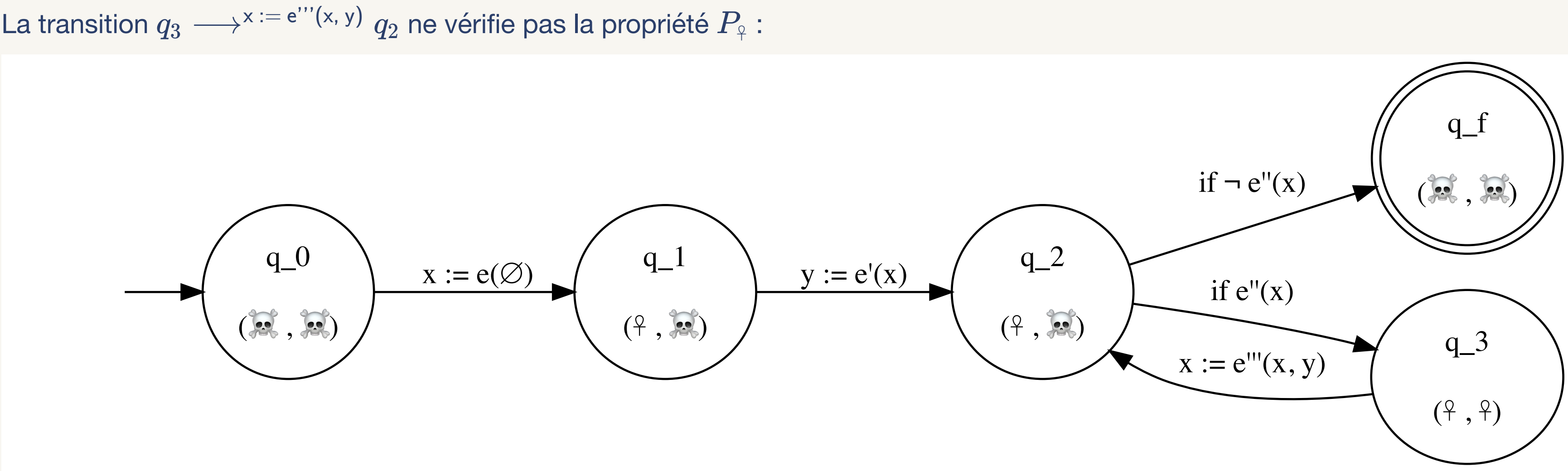
  - La transition $q_3 \longrightarrow^{x := e'''(x,\, y)} q_2$ ne vérifie pas la propriété $P_{♀}$:



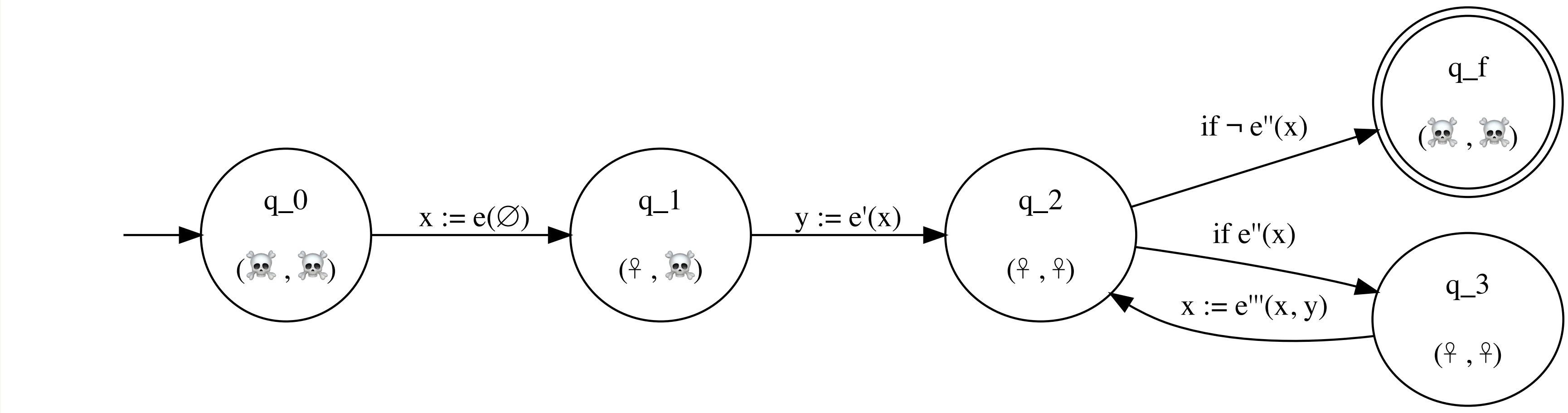
  - La transition $q_1 \longrightarrow^{y := e'(x)} q_2$ ne vérifie pas la propriété $P_{♀}$:



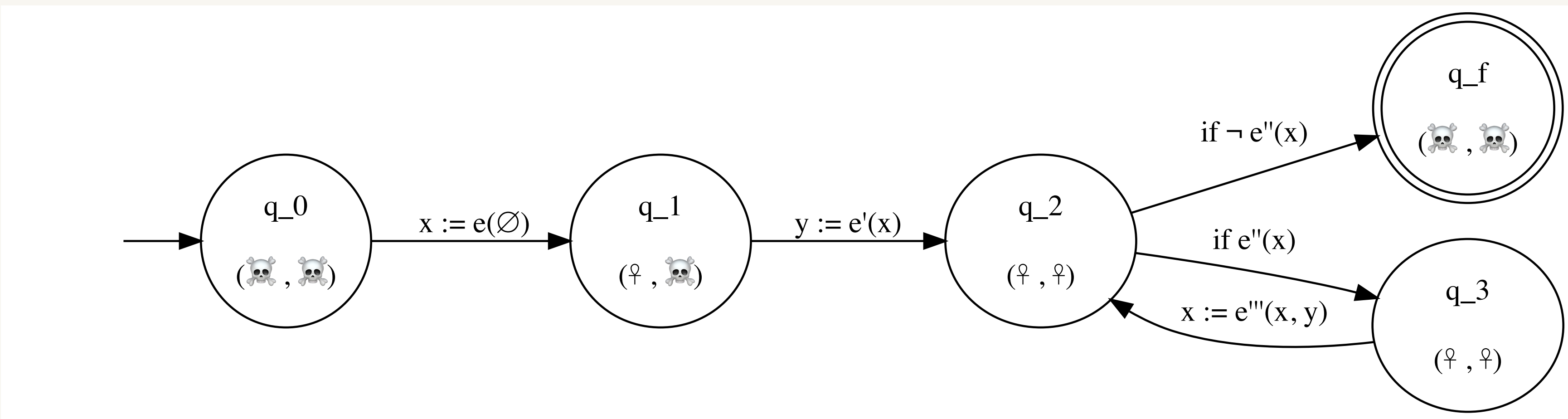
- $x_i = y$:
  - La transition $q_3 \longrightarrow^{x := e'''(x,\, y)} q_2$ ne vérifie pas la propriété $P_{♀}$:



  - La transition $q_2 \longrightarrow^{\text{if } e''(x)} q_3$ ne vérifie pas la propriété $P_{?}$:



- La transition $q_2 \longrightarrow^{\text{if e''(x)}} q_3$ ne vérifie pas la propriété $P_{\color{red}?}$:



Sortie :

q	$\Vert q \Vert$
q_0	(☠, ☠)
q_1	(♀, ☠)
q_2	(♀, ♀)
q_3	(♀, ♀)
q_f	(☠, ☠)

Propriétés de clôture

6.

Soient $f : \Sigma^\ast \longrightarrow S$ une fonction reconnue par un automate pondéré $\mathcal{A} \overset{\text{def}}{=} (Q, \Sigma, S, \lambda, \mu, \gamma)$, et $h : S \longrightarrow S'$ un morphisme.

Alors en posant :

$$\mathcal{A}_h \overset{\text{def}}{=} (Q, \Sigma, S, \lambda', \mu', \gamma')$$

- $\lambda' \overset{\text{def}}{=} h \circ \lambda$
- $\gamma' \overset{\text{def}}{=} h \circ \gamma$
- $\mu' \overset{\text{def}}{=} h \circ \mu$

il vient immédiatement (avec la notation matricielle) que pour tout mot w :

$$\begin{aligned} [\![\mathcal{A}_h]\!](w) &= \lambda' \mu'(w) \gamma' \\ &= (h \circ \lambda)(h \circ \mu)(w)(h \circ \gamma) \\ &= h \circ (\lambda \mu(w) \gamma) && \text{puisque } h \text{ respecte les lois de } S \\ &= h([\![\mathcal{A}]\!](w)) \\ &= h(f)(w) \end{aligned}$$

Donc

> si f est reconnaissable et si h est un morphisme, $h(f)$ est reconnaissable.

5.

Soient $f : \Sigma^\ast \longrightarrow S$ et $g : \Sigma^\ast \longrightarrow S$ deux fonctions reconnues respectivement par des automates pondérés $\mathcal{A}_f$ et $\mathcal{A}_g$.

Alors l'automate union $\mathcal{A}_{f+g}$ obtenu en juxtaposant (en mettant “côte à côte”) les automates $\mathcal{A}_f$ et $\mathcal{A}_g$ reconnaît $f + g$.

En effet, pour tout mot w (avec la notation matricielle et des notations évidentes) :

$$
\begin{aligned}
(f+g)(w) &= f(w)+g(w)\\
&= [\![\mathcal{A}_f]\!](w) + [\![\mathcal{A}_g]\!](w)\\
&= \lambda_f \mu_f(w)\gamma_f + \lambda_g \mu_g(w)\gamma_g
\end{aligned}
$$

et d'après la sémantique des automates pondérés, $\lambda_f \mu_f(w)\gamma_f + \lambda_g \mu_g(w)\gamma_g$ est le poids de w dans l'automate union (la somme des sommes des poids de tous les chemins étiquetés par w dans chacun des automates *égale* la somme des sommes des poids de tous les chemins étiquetés par w dans l'automate union, par définition de l'automate union).

Donc

> si f et g sont reconnaissables, $f+g$ l'est aussi.

6.

Soient $f : \Sigma^* \longrightarrow S$ et $g : \Sigma^* \longrightarrow S$ deux fonctions reconnues respectivement par des automates pondérés $\mathcal{A}_f$ et $\mathcal{A}_g$, et $(S, +, \times)$ un demi-anneau **commutatif**.

Rappels sur le produit de Kronecker $\otimes$:

Si $A \in \mathfrak{M}_{m,n}(S)$ et $B \in \mathfrak{M}_{p,q}(S)$:

$$
A \otimes B \overset{\text{def}}{=} \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}
$$

$$
= \begin{pmatrix}
a_{11}b_{11} & a_{11}b_{12} & \cdots & a_{11}b_{1q} & \cdots & \cdots & a_{1n}b_{11} & a_{1n}b_{12} & \cdots & a_{1n}b_{1q} \\
a_{11}b_{21} & a_{11}b_{22} & \cdots & a_{11}b_{2q} & \cdots & \cdots & a_{1n}b_{21} & a_{1n}b_{22} & \cdots & a_{1n}b_{2q} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
a_{11}b_{p1} & a_{11}b_{p2} & \cdots & a_{11}b_{pq} & \cdots & \cdots & a_{1n}b_{p1} & a_{1n}b_{p2} & \cdots & a_{1n}b_{pq} \\
\vdots & \vdots & & \vdots & \ddots & & \vdots & \vdots & & \vdots \\
\vdots & \vdots & & \vdots & & \ddots & \vdots & \vdots & & \vdots \\
a_{m1}b_{11} & a_{m1}b_{12} & \cdots & a_{m1}b_{1q} & \cdots & \cdots & a_{mn}b_{11} & a_{mn}b_{12} & \cdots & a_{mn}b_{1q} \\
a_{m1}b_{21} & a_{m1}b_{22} & \cdots & a_{m1}b_{2q} & \cdots & \cdots & a_{mn}b_{21} & a_{mn}b_{22} & \cdots & a_{mn}b_{2q} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
a_{m1}b_{p1} & a_{m1}b_{p2} & \cdots & a_{m1}b_{pq} & \cdots & \cdots & a_{mn}b_{p1} & a_{mn}b_{p2} & \cdots & a_{mn}b_{pq}
\end{pmatrix}
$$

On a montré en prépa (il suffit de faire le calcul) que :

> **Propriété du produit mixte** :
>
> Si A, B, C, D sont des matrices à coefficients dans un **demi-anneau commutatif**, sous réserve de compatibilité des tailles :
>
> $$(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$$

Montrons que $f \times g$ est reconnaissable.

On note n (resp. m) le nombre d'états de $\mathcal{A}_f$ (resp. $\mathcal{A}_g$).

Avec la notation matricielle (et des notations évidentes), pour tout mot w :

$$
\begin{aligned}
(f \times g)(w) &= \underbrace{f(w)}_{\in S} \times \underbrace{g(w)}_{\in S} \\
&= \underbrace{f(w)}_{\in S} \otimes \underbrace{g(w)}_{\in S} && \text{(les éléments de } S \text{ sont vus comme des matr} \\
&= (\lambda_f \mu_f(w)\gamma_f) \otimes (\lambda_g \mu_g(w)\gamma_g) \\
&= \left(\lambda_f \otimes \lambda_g\right)\left(\mu_f(w) \otimes \mu_g(w)\right)\left(\gamma_f \otimes \gamma_g\right) && \text{(produit mixte, car } S \text{ est commutatif)} \\
&= \begin{pmatrix} \lambda_{f,1}\lambda_{g,1} \\ \vdots \\ \lambda_{f,1}\lambda_{g,m} \\ \vdots \\ \lambda_{f,n}\lambda_{g,1} \\ \vdots \\ \lambda_{f,n}\lambda_{g,m} \end{pmatrix}
\begin{pmatrix} \mu_{f,11}(w)\mu_g(w) & \cdots & \mu_{f,1n}(w)\mu_g(w) \\ \vdots & \ddots & \vdots \\ \mu_{f,n1}(w)\mu_g(w) & \cdots & \mu_{f,nn}(w)\mu_g(w) \end{pmatrix}
\begin{pmatrix} \gamma_{f,1}\gamma_{g,1} \\ \vdots \\ \gamma_{f,1}\gamma_{g,m} \\ \vdots \\ \gamma_{f,n}\gamma_{g,1} \\ \vdots \\ \gamma_{f,n}\gamma_{g,m} \end{pmatrix} \\
&= [\![\mathcal{A}_{f\times g}]\!](w)
\end{aligned}
$$

en posant

$$\mathcal{A}_{f\times g} \overset{\text{def}}{=} (Q', \Sigma, S, \lambda', \mu', \gamma')$$

où

- $Q' \overset{\text{def}}{=} Q_f \times Q_g$
- $\forall q \in Q_f, q' \in Q_g, \quad \lambda'((q,q')) \overset{\text{def}}{=} \lambda_f(q) \times \lambda_g(q')$
- $\forall q \in Q_f, q' \in Q_g, \quad \gamma'((q,q')) \overset{\text{def}}{=} \gamma_f(q) \times \gamma_g(q')$
- $$\begin{aligned}&\forall a \in \Sigma, q_1, q_2 \in Q_f, q_1', q_2' \in Q_g,\\ &\quad \mu'((q_1,q_1'), a, (q_2,q_2')) \overset{\text{def}}{=} \mu_f(q_1,a,q_2) \times \mu_g(q_1',a,q_2')\end{aligned}$$

Donc

> si f et g sont reconnaissables et si S est commutatif, $f \times g$ est reconnaissable.

8.

L'hypothèse de commutativité de S est cruciale : le résultat précédent n'est pas vrai dans le cas général.

Contre-exemple

On se place sur le demi-anneau non commutatif $S \overset{\text{def}}{=} \langle \mathbb{P}(\Sigma^*), \cup, \cdot, \emptyset, \{\varepsilon\}\rangle$ et sur l'alphabet $\Sigma \overset{\text{def}}{=} \{a, b\}$.

La fonction f définie sur tout mot $w \in \Sigma^*$ par

$$f(w) \overset{\text{def}}{=} \{w\}$$

est clairement reconnaissable.

Mais **ce n'est pas le cas de** $f \times f \overset{\text{def}}{=} w \mapsto \{w\} \cdot \{w\} = \{w^2\}$.

Par l'absurde, si $\mathcal{A} \overset{\text{def}}{=} (Q, \Sigma, S, \lambda, \mu, \gamma)$ à $N \overset{\text{def}}{=} |Q|$ états reconnaissait $f \times f$: on obtiendrait une contradiction en imitant la démonstration du lemme de pompage pour les automates finis.

En effet : le chemin acceptant $q_{j_1} \longrightarrow^a q_{j_2} \longrightarrow^a \cdots \longrightarrow^b \underbrace{q_{j_m}}_{\in F}$ étiqueté par $a^{N+1}b$ (il est unique car $f^2(a^{N+1}b)$ est un singleton) passerait nécessairement deux fois par le même état $q_{j_k} = q_{j_{k'}}$ (disons que $k < k'$) lors de la lecture des $N+1$ premières lettres, et il existerait deux entiers $r \geq 1, l \geq 0$ tels que

- $k + l + r = N + 1$
- le même chemin où on a bouclé n fois sur $q_{j_k} \longrightarrow^a \cdots \longrightarrow^a q_{j_{k+r-1}} \longrightarrow^a q_{j_{k+r}} = q_{j_k}$ serait étiqueté par $a^k(a^r)^n a^l b$

Or :

- $f^2(a^{N+1}b) = \{a^{N+1}ba^{N+1}b\}$
- $f^2(a^k(a^r)^n a^l b) = \{a^k(a^r)^n a^l b a^k (a^r)^n a^l b\} = \{uv^n w\}$
  - où $uvw = a^{2(N+1)}$

on obtient un contradiction pour n assez grand :

- soit à cause du nombre de b si v contient la lettre b
- soit, si v ne contient pas la lettre b, en considérant les facteurs à gauche et à droite du b central (qui doivent être égaux dans $a^k(a^r)^n a^l b a^k (a^r)^n a^l b$, mais qui ne peuvent pas l'être dans $uv^n w$ si v ne contient pas la lettre b)

On a donc montré que :

> Si S n'est pas commutatif, et f et g sont reconnaissables, $f \times g$ n'est pas nécessairement reconnaissable.