

Théorie des Langages 1

Cours 5 : Expressions régulières

L. Rieg

Grenoble INP - Ensimag, 1^{re} année

Année 2023-2024

Définition

Définition (Expression régulière)

L'ensemble des **expressions régulières** (sur un vocabulaire V) est défini par induction structurelle par :

- **Base** :
 - ▶ \emptyset est une expression régulière
 - ▶ ϵ est une expression régulière
 - ▶ Si $x \in V$, alors x est une expression régulière

Définition

Définition (Expression régulière)

L'ensemble des **expressions régulières** (sur un vocabulaire V) est défini par induction structurelle par :

- **Base** :
 - ▶ \emptyset est une expression régulière
 - ▶ ϵ est une expression régulière
 - ▶ Si $x \in V$, alors x est une expression régulière
- **Induction** : Si E_1 et E_2 sont des expressions régulières, alors
 - ▶ $(E_1 + E_2)$ est une expression régulière
 - ▶ $(E_1.E_2)$ est une expression régulière
 - ▶ (E_1^*) est une expression régulière

Définition

Définition (Expression régulière)

L'ensemble des **expressions régulières** (sur un vocabulaire V) est défini par induction structurelle par :

- **Base** :
 - ▶ \emptyset est une expression régulière
 - ▶ ϵ est une expression régulière
 - ▶ Si $x \in V$, alors x est une expression régulière
- **Induction** : Si E_1 et E_2 sont des expressions régulières, alors
 - ▶ $(E_1 + E_2)$ est une expression régulière
 - ▶ $(E_1.E_2)$ est une expression régulière
 - ▶ (E_1^*) est une expression régulière

Exemple : $(a.((a + b)^*))$ est une expression régulière (ER).

Abréviations

- On pourra noter E_1E_2 à la place de $E_1.E_2$
- On pourra supprimer les parenthèses « inutiles » en éliminant la paire la plus externe et en considérant
 - ▶ « . » et « + » associatifs
 - ▶ « * » plus prioritaire que « . » plus prioritaire que « + »

Abréviations

- On pourra noter E_1E_2 à la place de $E_1.E_2$
- On pourra supprimer les parenthèses « inutiles » en éliminant la paire la plus externe et en considérant
 - ▶ « . » et « + » associatifs
 - ▶ « * » plus prioritaire que « . » plus prioritaire que « + »

Exemple

Soit $E = ((a.(b + (c + (d.(c^*))))).((a.b)^*))$.

On pourra simplement noter $E = a(b + c + dc^*)(ab)^*$.

Abréviations

- On pourra noter E_1E_2 à la place de $E_1.E_2$
- On pourra supprimer les parenthèses « inutiles » en éliminant la paire la plus externe et en considérant
 - ▶ « . » et « + » associatifs
 - ▶ « * » plus prioritaire que « . » plus prioritaire que « + »

Exemple

Soit $E = ((a.(b + (c + (d.(c^*))))).((a.b)^*))$.

On pourra simplement noter $E = a(b + c + dc^*)(ab)^*$.

Question : à quoi servent les ER ?

Abréviations

- On pourra noter E_1E_2 à la place de $E_1.E_2$
- On pourra supprimer les parenthèses « inutiles » en éliminant la paire la plus externe et en considérant
 - ▶ « . » et « + » associatifs
 - ▶ « * » plus prioritaire que « . » plus prioritaire que « + »

Exemple

Soit $E = ((a.(b + (c + (d.(c^*))))).((a.b)^*))$.

On pourra simplement noter $E = a(b + c + dc^*)(ab)^*$.

Question : à quoi servent les ER ?

- Intérêt pratique : **recherche de motif** (cf. grep et sed)
- Intérêt théorique : description inductive des langages réguliers

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon,), (, +, *, \cdot\}$ qui **représente un langage** sur V .

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon,), (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cup \mathcal{L}(E_2)$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cup \mathcal{L}(E_2)$
- Si $E = (E_1.E_2)$ alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cup \mathcal{L}(E_2)$
- Si $E = (E_1.E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1).\mathcal{L}(E_2)$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cup \mathcal{L}(E_2)$
- Si $E = (E_1 \cdot E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cdot \mathcal{L}(E_2)$
- Si $E = (E_1^*)$ alors $\mathcal{L}(E) =$

Langage représenté

Une expression régulière sur V est un mot sur $V \cup \{\emptyset, \epsilon, \cdot, (, +, *, \cdot\}$ qui **représente un langage** sur V .

Définition

Le **langage représenté** par une ER E est noté $\mathcal{L}(E)$ et est défini par :

- Si $E = \emptyset$ alors $\mathcal{L}(E) = \emptyset$
- Si $E = \epsilon$ alors $\mathcal{L}(E) = \{\epsilon\}$
- Si $E = x$ ($x \in V$) alors $\mathcal{L}(E) = \{x\}$
- Si $E = (E_1 + E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cup \mathcal{L}(E_2)$
- Si $E = (E_1 \cdot E_2)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1) \cdot \mathcal{L}(E_2)$
- Si $E = (E_1^*)$ alors $\mathcal{L}(E) = \mathcal{L}(E_1)^*$

Remarques

Exemple

$$\mathcal{L}((a + ab)^*) =$$

Remarques

Exemple

$$\mathcal{L}((a + ab)^*) = (\mathcal{L}(a + ab))^*$$

Remarques

Exemple

$$\mathcal{L}((a + ab)^*) = (\mathcal{L}(a + ab))^* = (\mathcal{L}(a) \cup \mathcal{L}(ab))^*$$

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* = (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^*\end{aligned}$$

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* &= (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^* &= (\{a\} \cup \{a\}.\{b\})^*\end{aligned}$$

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* &= (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^* &= (\{a\} \cup \{a\}.\{b\})^* \\ &= \{a, ab\}^*\end{aligned}$$

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* &= (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^* &= (\{a\} \cup \{a\}.\{b\})^* \\ &= \{a, ab\}^*\end{aligned}$$

Remarques

- *L'associativité dans les ER (pour « . » et « + ») vient des langages.*
- *Priorités comme en arithmétique : « * » > « . » > « + »*

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* &= (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^* &= (\{a\} \cup \{a\}.\{b\})^* \\ &= \{a, ab\}^*\end{aligned}$$

Remarques

- *L'associativité dans les ER (pour « . » et « + ») vient des langages.*
- *Priorités comme en arithmétique : « * » > « . » > « + »*

Notations

- *On pourra noter E à la place de $\mathcal{L}(E)$.*
- *Du coup, des notations comme $w \in E$ ou $A \subseteq B + C$ sont autorisées.*

Remarques

Exemple

$$\begin{aligned}\mathcal{L}((a + ab)^*) &= (\mathcal{L}(a + ab))^* &= (\mathcal{L}(a) \cup \mathcal{L}(ab))^* \\ &= (\mathcal{L}(a) \cup \mathcal{L}(a).\mathcal{L}(b))^* &= (\{a\} \cup \{a\}.\{b\})^* \\ &= \{a, ab\}^*\end{aligned}$$

Remarques

- *L'associativité dans les ER (pour « . » et « + ») vient des langages.*
- *Priorités comme en arithmétique : « * » > « . » > « + »*

Notations

- *On pourra noter E à la place de $\mathcal{L}(E)$.*
- *Du coup, des notations comme $w \in E$ ou $A \subseteq B + C$ sont autorisées.*
- *Il ne faut pas tout mélanger : on n'écrira pas $w \in (a + b)^* \{c, d\}$.*

Expressions régulières équivalentes

Définition

Deux expressions régulières E et E' sont **équivalentes** si $\mathcal{L}(E) = \mathcal{L}(E')$.

Expressions régulières équivalentes

Définition

Deux expressions régulières E et E' sont **équivalentes** si $\mathcal{L}(E) = \mathcal{L}(E')$.

Exemple

Les expressions régulières $(a + b)^*$ et $(a^*b^*)^*$ sont équivalentes.

Exercice : démontrer ce résultat

Expressions régulières équivalentes

Définition

Deux expressions régulières E et E' sont **équivalentes** si $\mathcal{L}(E) = \mathcal{L}(E')$.

Exemple

Les expressions régulières $(a + b)^*$ et $(a^*b^*)^*$ sont équivalentes.

Exercice : démontrer ce résultat

Théorème (Kleene)

Les langages représentés par des expressions régulières sont les langages réguliers.

Expressions régulières équivalentes

Définition

Deux expressions régulières E et E' sont **équivalentes** si $\mathcal{L}(E) = \mathcal{L}(E')$.

Exemple

Les expressions régulières $(a + b)^*$ et $(a^*b^*)^*$ sont équivalentes.

Exercice : démontrer ce résultat

Théorème (Kleene)

Les langages représentés par des expressions régulières sont les langages réguliers.

Rappel : L régulier $\stackrel{\text{def}}{=} \exists A(\text{AF}) : \mathcal{L}(A) = L$

Expressions régulières équivalentes

Définition

Deux expressions régulières E et E' sont **équivalentes** si $\mathcal{L}(E) = \mathcal{L}(E')$.

Exemple

Les expressions régulières $(a + b)^*$ et $(a^*b^*)^*$ sont équivalentes.

Exercice : démontrer ce résultat

Théorème (Kleene)

Les langages représentés par des expressions régulières sont les langages réguliers.

Rappel : L régulier $\stackrel{\text{def}}{=} \exists A(\text{AF}) : \mathcal{L}(A) = L$

- À démontrer :
1. $\forall E(\text{ER}), \exists A(\text{AF}) : \mathcal{L}(E) = \mathcal{L}(A)$
 2. $\forall A(\text{AF}), \exists E(\text{ER}) : \mathcal{L}(A) = \mathcal{L}(E)$

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

De plus, A a un unique état initial et un unique état acceptant.

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

- $E = \emptyset$

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

- $E = \emptyset$



Une ER représente un langage régulier

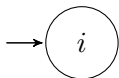
Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

• $E = \emptyset$



• $E = \epsilon$



Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

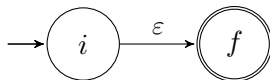
De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

• $E = \emptyset$



• $E = \epsilon$



Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

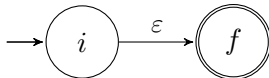
De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

- $E = \emptyset$



- $E = \epsilon$



- $E = x \in V$

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

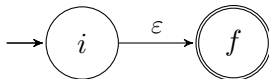
De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

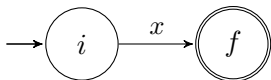
• $E = \emptyset$



• $E = \epsilon$



• $E = x \in V$



Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

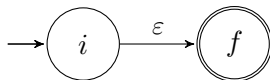
De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

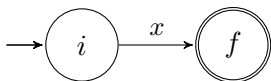
- $E = \emptyset$



- $E = \epsilon$



- $E = x \in V$



- $E = (E_1 + E_2), (E_1.E_2), (E_1^*)$

Une ER représente un langage régulier

Lemme

$\forall E(ER), \exists A(AF)$ tel que $\mathcal{L}(E) = \mathcal{L}(A)$

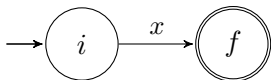
De plus, A a un unique état initial et un unique état acceptant.

Preuve : par induction structurelle

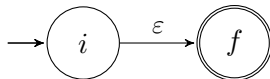
- $E = \emptyset$



- $E = x \in V$



- $E = \epsilon$



- $E = (E_1 + E_2), (E_1.E_2), (E_1^*)$

Hyp. Ind. + voir le cours 3

Un langage régulier est représentable par une ER

2 visions possibles :

- Version graphique :

- Version par équations :

Un langage régulier est représentable par une ER

2 visions possibles :

- Version graphique :

- ▶ **Idée** : se ramener à un automate de la forme



- Version par équations :

Un langage régulier est représentable par une ER

2 visions possibles :

- Version graphique :

- ▶ **Idée** : se ramener à un automate de la forme



- ▶ Méthode :

- ★ S'autoriser **des ER sur les transitions**
- ★ Partir d'un automate avec un unique état initial i sans transition entrante et un unique état acceptant f sans transition sortante (voir cours 3)
- ★ **Supprimer successivement les états** (sauf i et f) en préservant le langage reconnu

- Version par équations :

Un langage régulier est représentable par une ER

2 visions possibles :

- Version graphique :

- ▶ **Idée** : se ramener à un automate de la forme



- ▶ Méthode :

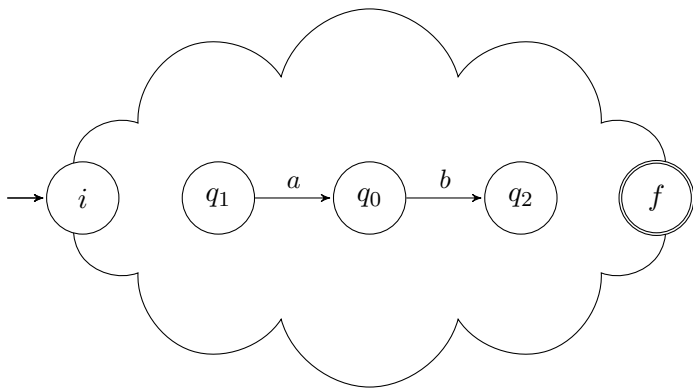
- ★ S'autoriser **des ER sur les transitions**
 - ★ Partir d'un automate avec un unique état initial i sans transition entrante et un unique état acceptant f sans transition sortante (voir cours 3)
 - ★ **Supprimer successivement les états** (sauf i et f) en préservant le langage reconnu
- Version par équations : **résolution d'un système d'équations d'ER**
 - ▶ Pas besoin de transformer l'automate
 - ▶ Représentation algébrique de la version graphique

Version graphique

Comment supprimer un état d'un automate sans changer son langage ?

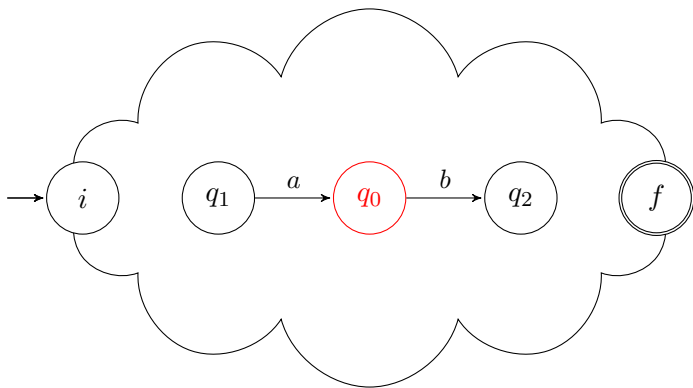
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



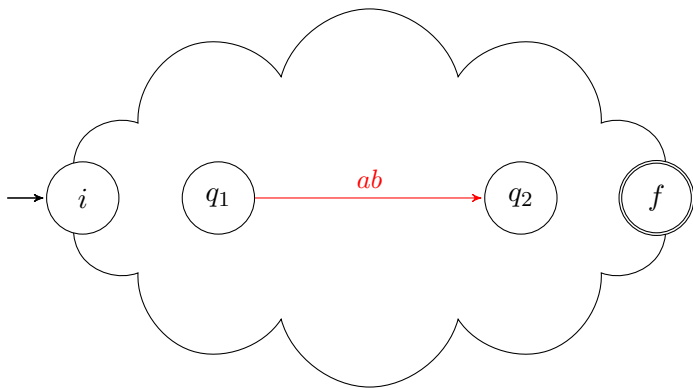
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



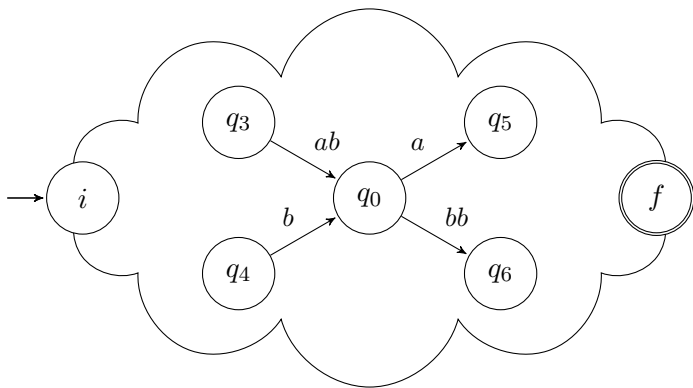
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



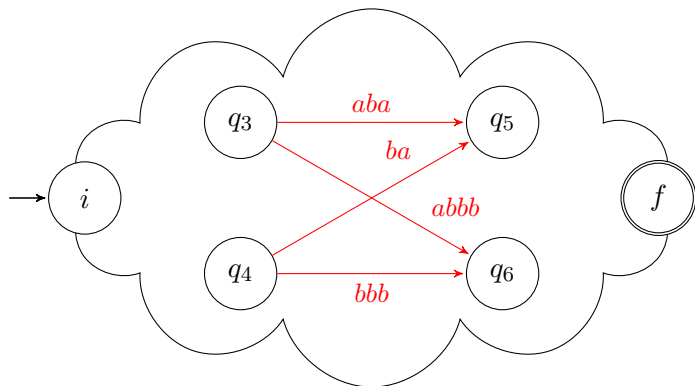
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



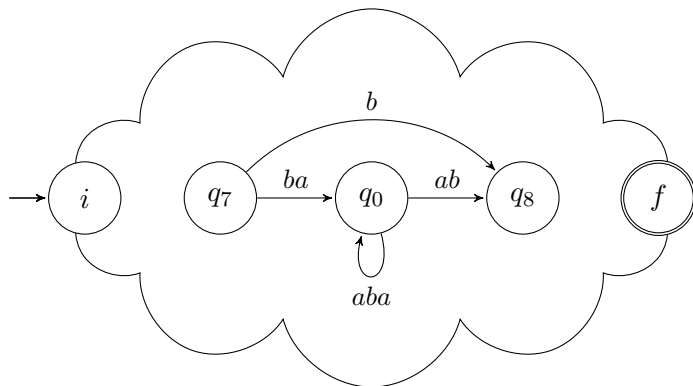
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



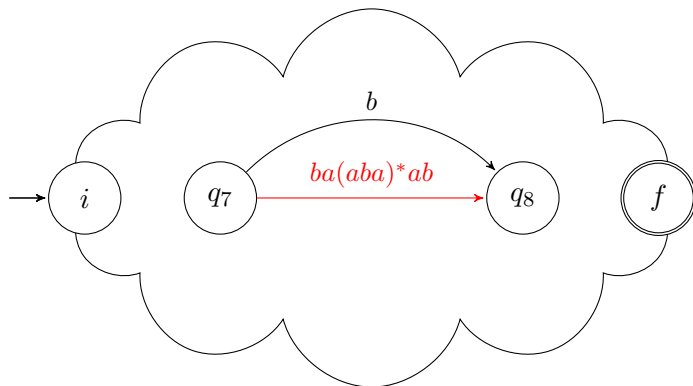
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



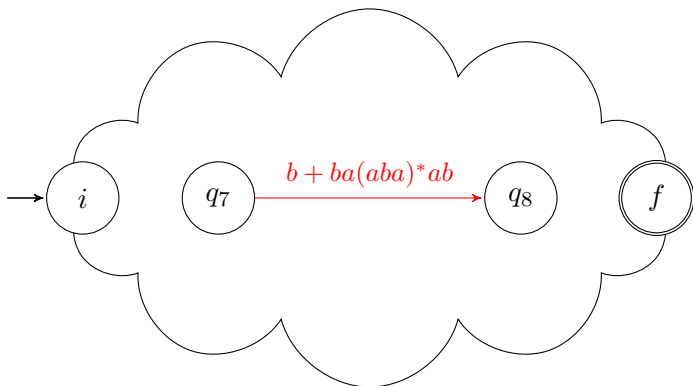
Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



Version graphique

Comment supprimer un état d'un automate sans changer son langage ?



Version graphique

Comment supprimer un état d'un automate sans changer son langage ?

Formellement, pour supprimer un état q avec

- des transitions entrantes $(p, x, q) \in \delta$ (avec $p \neq q$)
- possiblement une boucle $(q, y, q) \in \delta$
- des transitions sortantes $(q, z, r) \in \delta$ (avec $r \neq q$)

on doit

Version graphique

Comment supprimer un état d'un automate sans changer son langage ?

Formellement, pour supprimer un état q avec

- des transitions entrantes $(p, x, q) \in \delta$ (avec $p \neq q$)
- possiblement une boucle $(q, y, q) \in \delta$
- des transitions sortantes $(q, z, r) \in \delta$ (avec $r \neq q$)

on doit

- supprimer l'état q et les transitions ci-dessus,
- ajouter à δ les transitions (p, xy^*z, r) (pour chaque couple (p, r)),
- possiblement fusionner des transitions parallèles.

Version graphique

Comment supprimer un état d'un automate sans changer son langage ?

Formellement, pour supprimer un état q avec

- des transitions entrantes $(p, x, q) \in \delta$ (avec $p \neq q$)
- possiblement une boucle $(q, y, q) \in \delta$
- des transitions sortantes $(q, z, r) \in \delta$ (avec $r \neq q$)

on doit

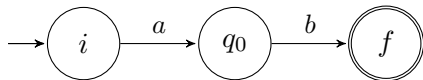
- supprimer l'état q et les transitions ci-dessus,
- ajouter à δ les transitions (p, xy^*z, r) (pour chaque couple (p, r)),
- possiblement fusionner des transitions parallèles.

C'est la **méthode de Brzowski et Mc Cuskey**.

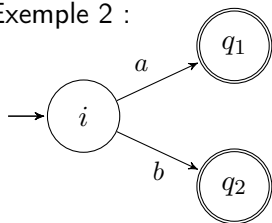
On peut programmer cette méthode : l'**algorithme de Kleene** (prog. dyn.)

Version graphique : exemples

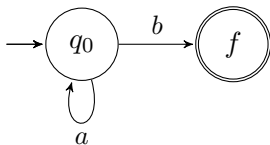
- Exemple 1 :



- Exemple 2 :

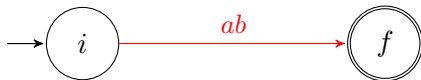


- Exemple 3 :

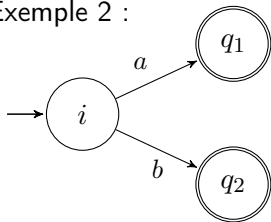


Version graphique : exemples

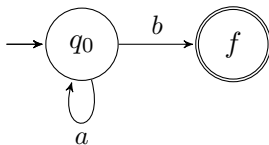
- Exemple 1 :



- Exemple 2 :

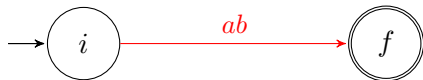


- Exemple 3 :

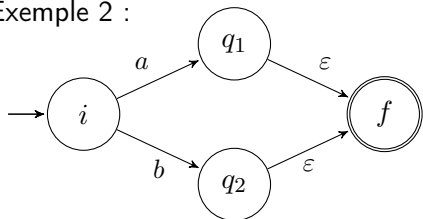


Version graphique : exemples

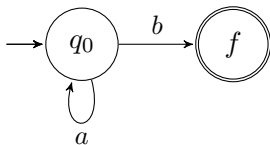
- Exemple 1 :



- Exemple 2 :

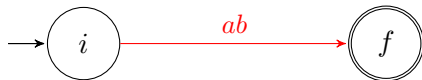


- Exemple 3 :

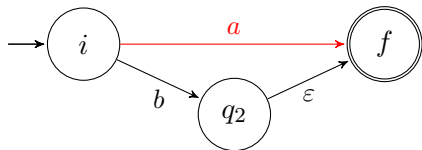


Version graphique : exemples

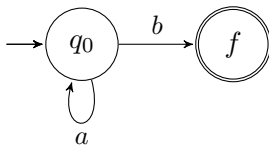
- Exemple 1 :



- Exemple 2 :

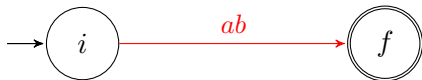


- Exemple 3 :

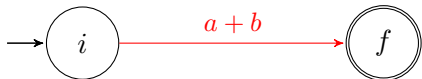


Version graphique : exemples

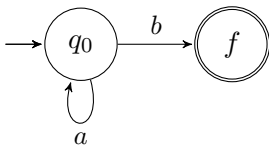
- Exemple 1 :



- Exemple 2 :

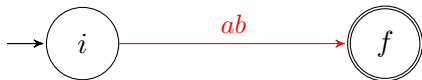


- Exemple 3 :

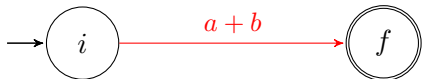


Version graphique : exemples

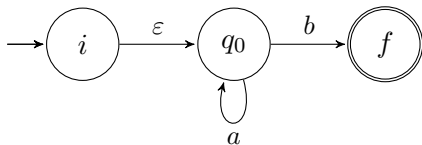
- Exemple 1 :



- Exemple 2 :

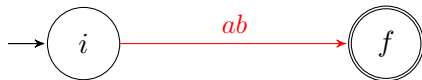


- Exemple 3 :

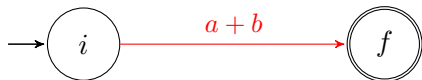


Version graphique : exemples

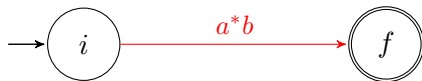
- Exemple 1 :



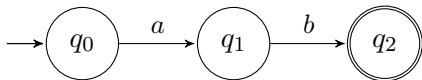
- Exemple 2 :



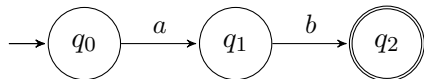
- Exemple 3 :



Version par équations : exemple 1



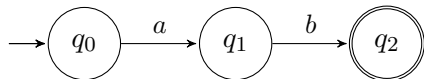
Version par équations : exemple 1



Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a
$$\mathcal{L}(A) = \mathcal{L}(A_{q_0})$$

Version par équations : exemple 1

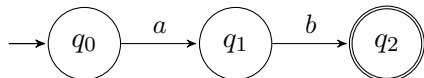


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a $\mathcal{L}(A) = \mathcal{L}(A_{q_0})$

$$\left\{ \begin{array}{l} \mathcal{L}(A_{q_0}) = \{a\} . \mathcal{L}(A_{q_1}) \\ \end{array} \right.$$

Version par équations : exemple 1

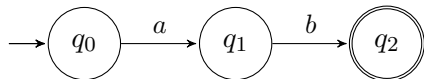


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a
$$\mathcal{L}(A) = \mathcal{L}(A_{q_0})$$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \end{cases}$$

Version par équations : exemple 1

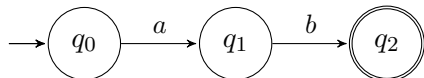


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a
$$\mathcal{L}(A) = \mathcal{L}(A_{q_0})$$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 1

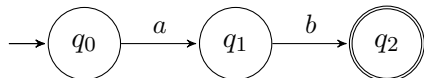


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a $\mathcal{L}(A) = \mathcal{L}(A_{q_0})$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases} \implies \begin{cases} \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 1

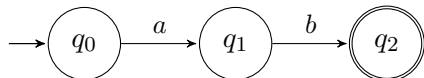


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a
$$\mathcal{L}(A) = \mathcal{L}(A_{q_0})$$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases} \implies \begin{cases} \mathcal{L}(A_{q_1}) = \{b\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 1

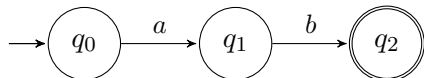


Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a $\mathcal{L}(A) = \mathcal{L}(A_{q_0})$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases} \implies \begin{cases} \mathcal{L}(A_{q_0}) = \{ab\} \\ \mathcal{L}(A_{q_1}) = \{b\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 1



Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

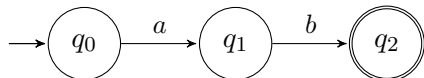
On a $\mathcal{L}(A) = \mathcal{L}(A_{q_0})$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases} \implies \begin{cases} \mathcal{L}(A_{q_0}) = \{ab\} \\ \mathcal{L}(A_{q_1}) = \{b\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Système d'équations (sur des langages / des ER) :

$$\begin{cases} x_0 = ax_1 \\ x_1 = bx_2 \\ x_2 = \epsilon \end{cases}$$

Version par équations : exemple 1



Considérons A_{q_i} la variante de A où l'unique état initial est q_i .

On a $\mathcal{L}(A) = \mathcal{L}(A_{q_0})$

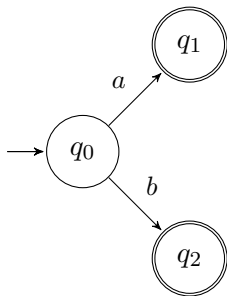
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases} \implies \begin{cases} \mathcal{L}(A_{q_0}) = \{ab\} \\ \mathcal{L}(A_{q_1}) = \{b\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Système d'équations (sur des langages / des ER) :

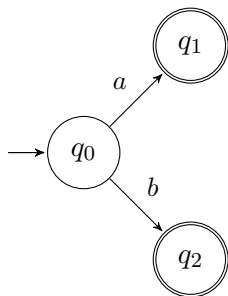
$$\begin{cases} x_0 &= ax_1 \\ x_1 &= bx_2 \\ x_2 &= \epsilon \end{cases}$$

Solution : $x_0 = ab$

Version par équations : exemple 2

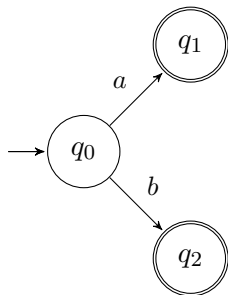


Version par équations : exemple 2



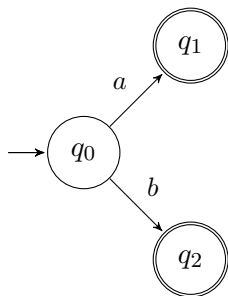
$$\left\{ \begin{array}{l} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \end{array} \right.$$

Version par équations : exemple 2



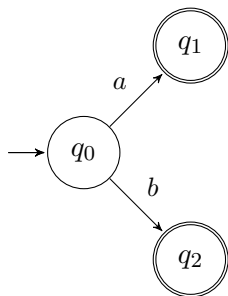
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 2



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

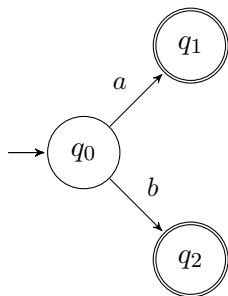
Version par équations : exemple 2



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

$$\begin{cases} \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

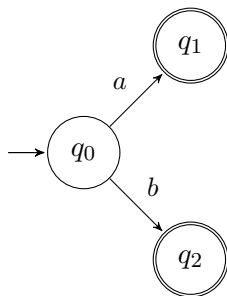
Version par équations : exemple 2



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

$$\begin{cases} \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

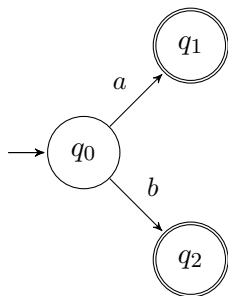
Version par équations : exemple 2



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a, b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 2



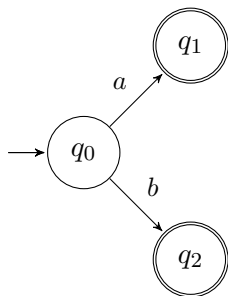
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a, b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Systeme d'equations :

$$\begin{cases} x_0 = ax_1 + bx_2 \\ x_1 = \epsilon \\ x_2 = \epsilon \end{cases}$$

Version par équations : exemple 2



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_1}) \cup \{b\} \cdot \mathcal{L}(A_{q_2}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

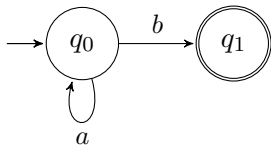
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a, b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \\ \mathcal{L}(A_{q_2}) = \{\varepsilon\} \end{cases}$$

Système d'équations :

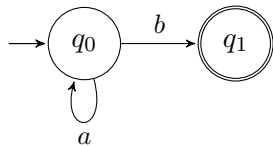
$$\begin{cases} x_0 = ax_1 + bx_2 \\ x_1 = \epsilon \\ x_2 = \epsilon \end{cases}$$

Solution : $x_0 = a + b$

Version par équations : exemple 3

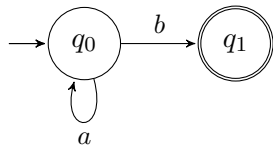


Version par équations : exemple 3



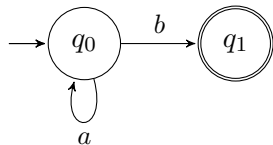
$$\left\{ \begin{array}{l} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \end{array} \right.$$

Version par équations : exemple 3



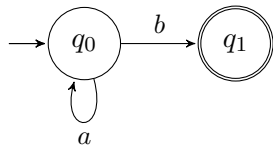
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 3



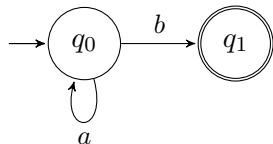
$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases} \quad \Longrightarrow \quad \begin{cases} \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

Version par équations : exemple 3



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases} \quad \Longrightarrow \quad \begin{cases} \mathcal{L}(A_{q_0}) = \{a\}^* \{b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

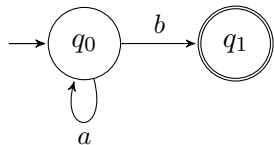
Version par équations : exemple 3



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases} \quad \Longrightarrow \quad \begin{cases} \mathcal{L}(A_{q_0}) = \{a\}^* \{b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

$$\text{Système d'équations : } \begin{cases} x_0 = ax_0 + bx_1 \\ x_1 = \epsilon \end{cases}$$

Version par équations : exemple 3



$$\begin{cases} \mathcal{L}(A_{q_0}) = \{a\} \cdot \mathcal{L}(A_{q_0}) \cup \{b\} \cdot \mathcal{L}(A_{q_1}) \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases} \quad \Longrightarrow \quad \begin{cases} \mathcal{L}(A_{q_0}) = \{a\}^* \{b\} \\ \mathcal{L}(A_{q_1}) = \{\varepsilon\} \end{cases}$$

$$\text{Système d'équations : } \begin{cases} x_0 = ax_0 + bx_1 \\ x_1 = \epsilon \end{cases}$$

$$\text{Solution : } x_0 = a^*b$$

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i
 - ▶ $(q_i, a_{i_1}, q_{i_1}), (q_i, a_{i_2}, q_{i_2}), \dots, (q_i, a_{i_k}, q_{i_k})$ où $a_{i_j} \in V \cup \{\varepsilon\}$

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i
 - ▶ $(q_i, a_{i_1}, q_{i_1}), (q_i, a_{i_2}, q_{i_2}), \dots, (q_i, a_{i_k}, q_{i_k})$ où $a_{i_j} \in V \cup \{\varepsilon\}$
 - ▶ On crée l'équation :

$$x_i = \sum_{j=1}^k a_{i_j} x_{i_j}$$

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i
 - ▶ $(q_i, a_{i_1}, q_{i_1}), (q_i, a_{i_2}, q_{i_2}), \dots, (q_i, a_{i_k}, q_{i_k})$ où $a_{i_j} \in V \cup \{\varepsilon\}$
 - ▶ On crée l'équation :

$$x_i = \sum_{j=1}^k a_{i_j} x_{i_j} + \epsilon \text{ si } q_i \text{ est final}$$

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i
 - ▶ $(q_i, a_{i_1}, q_{i_1}), (q_i, a_{i_2}, q_{i_2}), \dots, (q_i, a_{i_k}, q_{i_k})$ où $a_{i_j} \in V \cup \{\varepsilon\}$
 - ▶ On crée l'équation :

$$x_i = \sum_{j=1}^k a_{i_j} x_{i_j} + \epsilon \text{ si } q_i \text{ est final}$$

Remarque : Si $k = 0$, la somme se réduit à \emptyset et on a alors
 $x_i = \emptyset$ si $q_i \notin F$ ou $x_i = \epsilon$ si $q_i \in F$

Système d'équations associé à un automate A

Pour chaque état q_i :

- On considère une variable x_i , qui représentera le langage reconnu par l'automate dont q_i est l'unique état initial (A_{q_i})
- On considère les k transitions issues de q_i
 - ▶ $(q_i, a_{i_1}, q_{i_1}), (q_i, a_{i_2}, q_{i_2}), \dots, (q_i, a_{i_k}, q_{i_k})$ où $a_{i_j} \in V \cup \{\epsilon\}$
 - ▶ On crée l'équation :

$$x_i = \sum_{j=1}^k a_{i_j} x_{i_j} + \epsilon \text{ si } q_i \text{ est final}$$

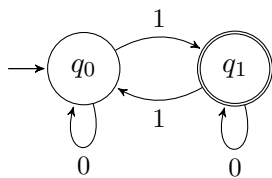
Remarque : Si $k = 0$, la somme se réduit à \emptyset et on a alors

$$x_i = \emptyset \text{ si } q_i \notin F \quad \text{ou} \quad x_i = \epsilon \text{ si } q_i \in F$$

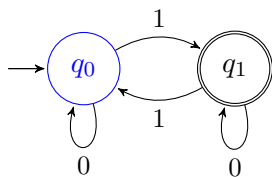
Théorème (Admis)

Pour tout i , la **plus petite** solution de l'équation associée à q_i représente le langage reconnu par l'automate dont q_i est l'unique état initial.

Exemple

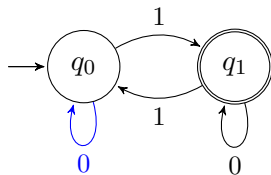


Exemple



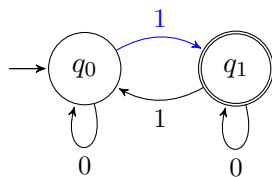
$$\left\{ x_0 = \right.$$

Exemple



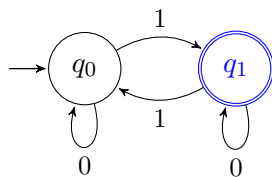
$$\left\{ \begin{array}{l} x_0 = 0x_0 \end{array} \right.$$

Exemple



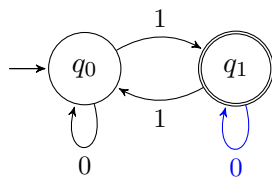
$$\left\{ \begin{array}{l} x_0 = 0x_0 + 1x_1 \end{array} \right.$$

Exemple



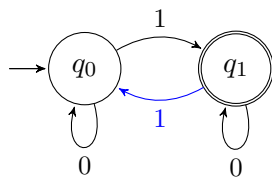
$$\begin{cases} x_0 & = & 0x_0 + 1x_1 \\ x_1 & = & \end{cases}$$

Exemple



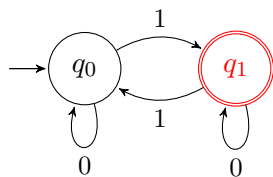
$$\begin{cases} x_0 &= 0x_0 + 1x_1 \\ x_1 &= 0x_1 \end{cases}$$

Exemple



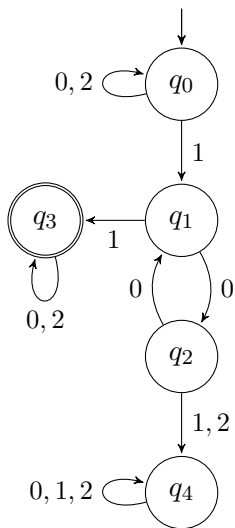
$$\begin{cases} x_0 &= 0x_0 + 1x_1 \\ x_1 &= 0x_1 + 1x_0 \end{cases}$$

Exemple

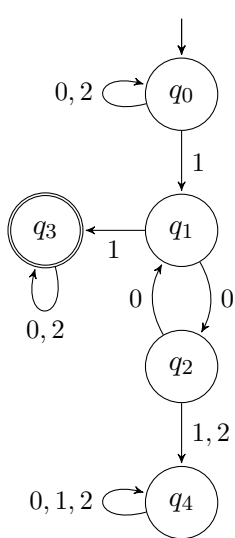


$$\begin{cases} x_0 &= 0x_0 + 1x_1 \\ x_1 &= 0x_1 + 1x_0 + \epsilon \end{cases}$$

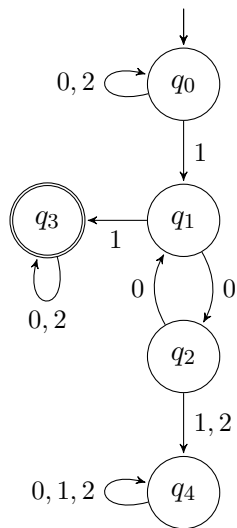
Exercise



Exercise

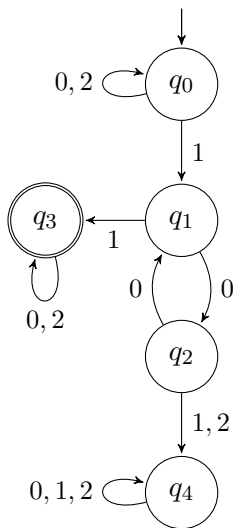


Exercise



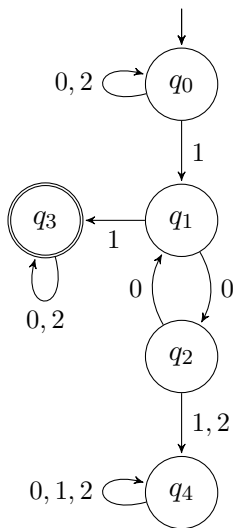
$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \end{array} \right.$$

Exercise



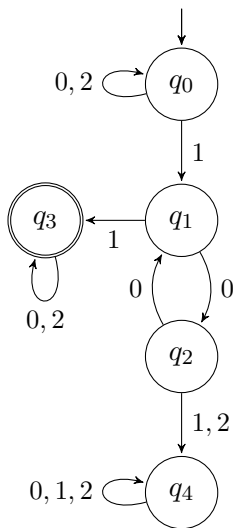
$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \end{array} \right.$$

Exercise



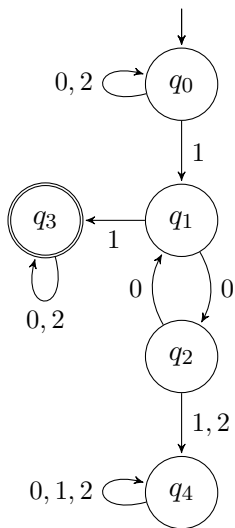
$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \end{array} \right.$$

Exercise



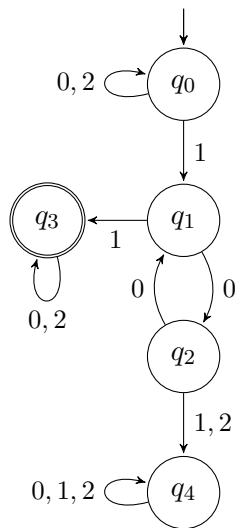
$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 \end{array} \right.$$

Exercise



$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \end{array} \right.$$

Exercise



$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\varepsilon \notin A$, c'est l'unique solution

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\varepsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\varepsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

$$A(A^*B) + B$$

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\varepsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

$$A(A^*B) + B = A^+B + B$$

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\epsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

$$\begin{aligned} A(A^*B) + B &= A^+B + B \\ &= A^+B + \epsilon B \end{aligned}$$

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\epsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

$$\begin{aligned}A(A^*B) + B &= A^+B + B \\ &= A^+B + \epsilon B \\ &= (A^+ + \epsilon)B\end{aligned}$$

Résolution des systèmes d'équations

Lemme d'Arden

Soient A et B des langages, et considérons l'équation $X = AX + B$.

Alors :

- A^*B est la plus petite solution de cette équation
- Si $\epsilon \notin A$, c'est l'unique solution

Preuve

- A^*B est une solution

$$\begin{aligned}A(A^*B) + B &= A^+B + B \\ &= A^+B + \epsilon B \\ &= (A^+ + \epsilon)B \\ &= A^*B\end{aligned}$$

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Donc $AB \subseteq AC \subseteq C$

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Donc $AB \subseteq AC \subseteq C$

Donc $A^2B \subseteq AC \subseteq C$

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Donc $AB \subseteq AC \subseteq C$

Donc $A^2B \subseteq AC \subseteq C$

\vdots

Donc $\forall k \geq 0, A^k B \subseteq C$

Preuve par récurrence sur k (exercice)

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Donc $AB \subseteq AC \subseteq C$

Donc $A^2B \subseteq AC \subseteq C$

⋮

Donc $\forall k \geq 0, A^k B \subseteq C$

Donc $\bigcup_{k \geq 0} A^k B \subseteq C$

Preuve par récurrence sur k (exercice)

Résolution des systèmes d'équations

Preuve (suite)

- A^*B est la plus petite solution

Soit C une solution : on a $C = AC + B$.

Donc $B \subseteq C$ et $AC \subseteq C$

Donc $AB \subseteq AC \subseteq C$

Donc $A^2B \subseteq AC \subseteq C$

\vdots

Preuve par récurrence sur k (exercice)

Donc $\forall k \geq 0, A^k B \subseteq C$

Donc $\bigcup_{k \geq 0} A^k B \subseteq C$

Or $\bigcup_{k \geq 0} A^k B = \left(\bigcup_{k \geq 0} A^k \right) B = A^* B$

Ainsi $A^* B \subseteq C$

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale**

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

- ▶ Si $w \in B$ alors $w \in A^*B$, **absurde**

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

- ▶ Si $w \in B$ alors $w \in A^*B$, **absurde**
- ▶ Sinon $w = w_1w_2$, où $w_1 \in A$ et $w_2 \in C$

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

- ▶ Si $w \in B$ alors $w \in A^*B$, **absurde**
- ▶ Sinon $w = w_1w_2$, où $w_1 \in A$ et $w_2 \in C$
Par hypothèse, $w_1 \neq \varepsilon$ (car $\varepsilon \notin A$), donc $|w_2| < |w|$

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

- ▶ Si $w \in B$ alors $w \in A^*B$, **absurde**
- ▶ Sinon $w = w_1w_2$, où $w_1 \in A$ et $w_2 \in C$
Par hypothèse, $w_1 \neq \varepsilon$ (car $\varepsilon \notin A$), donc $|w_2| < |w|$, d'où $w_2 \in A^*B$.

Résolution des systèmes d'équations

Preuve (suite)

- Si $\varepsilon \notin A$ alors A^*B est l'unique solution.

Soit C une solution, montrons que $C = A^*B$.

Comme A^*B est la plus petite solution, on a $A^*B \subseteq C$. Il suffit donc de montrer que $C \subseteq A^*B$.

Par l'absurde : on suppose que $\exists w \in C$ tel que $w \notin A^*B$.

Supposons w de longueur **minimale** :

si $w' \in C$ et $|w'| < |w|$ alors $w' \in A^*B$.

Par hypothèse, $C = AC + B$, donc $w \in B$ ou $w \in AC$.

- ▶ Si $w \in B$ alors $w \in A^*B$, **absurde**
- ▶ Sinon $w = w_1w_2$, où $w_1 \in A$ et $w_2 \in C$
Par hypothèse, $w_1 \neq \varepsilon$ (car $\varepsilon \notin A$), donc $|w_2| < |w|$, d'où $w_2 \in A^*B$.
Mais dans ce cas, on a $w = w_1w_2 \in AA^*B \subseteq A^*B$

Contradiction

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?
 - ▶ Renvoyer $Sol(x_j) + Sol(x_k)$.

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?
 - ▶ Renvoyer $Sol(x_j) + Sol(x_k)$.
- Si $\epsilon \in A$, quelles autres solutions de $X = AX + B$ y a-t-il?

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?
 - ▶ Renvoyer $Sol(x_j) + Sol(x_k)$.
- Si $\epsilon \in A$, quelles autres solutions de $X = AX + B$ y a-t-il?
 - ▶ N'importe quel X de la forme A^*C avec $B \subseteq C$ est solution.

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?
 - ▶ Renvoyer $Sol(x_j) + Sol(x_k)$.
- Si $\epsilon \in A$, quelles autres solutions de $X = AX + B$ y a-t-il?
 - ▶ N'importe quel X de la forme A^*C avec $B \subseteq C$ est solution.
- Qu'obtient-on pour l'équation $X = XA + B$?

Remarques importantes

Questions

- Quelle est la (plus petite) solution de l'équation $X = AX + \epsilon$?
 - ▶ $X = A^*$
- Quelle est la (plus petite) solution de l'équation $X = AX$?
 - ▶ $X = \emptyset$
- Si l'automate a deux états initiaux q_j et q_k ?
 - ▶ Renvoyer $Sol(x_j) + Sol(x_k)$.
- Si $\epsilon \in A$, quelles autres solutions de $X = AX + B$ y a-t-il?
 - ▶ N'importe quel X de la forme A^*C avec $B \subseteq C$ est solution.
- Qu'obtient-on pour l'équation $X = XA + B$?
 - ▶ La plus petite solution est BA^* .

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} \\ \\ \\ \\ x_4 = \emptyset \end{array} \right.$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} \\ \\ \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} x_1 = 00x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 00x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{cases}$$

$$\begin{cases} x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{cases}$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 00x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{cases}$$

$$\begin{cases} x_1 = (00)^*1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{cases}$$

Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 00x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

$$\left\{ \begin{array}{l} x_0 = (0 + 2)^*1(00)^*1(0 + 2)^* \\ x_1 = (00)^*1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

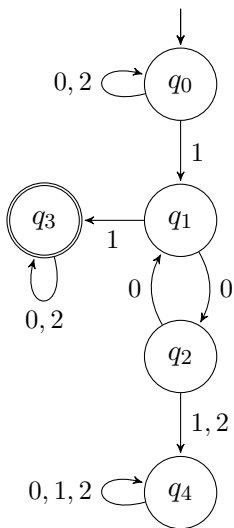
Exercice

Résoudre le système d'équations suivant (q_0 état initial) :

$$\left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{array} \right. \quad \left\{ \begin{array}{l} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 00x_1 + 1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right.$$

$$\left\{ \begin{array}{l} x_0 = (0 + 2)^*1(00)^*1(0 + 2)^* \\ x_1 = (00)^*1(0 + 2)^* \\ x_2 = 0x_1 \\ x_3 = (0 + 2)^* \\ x_4 = \emptyset \end{array} \right. \quad \text{donc } \mathcal{L}(A) = \mathcal{L}(A_{q_0}) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

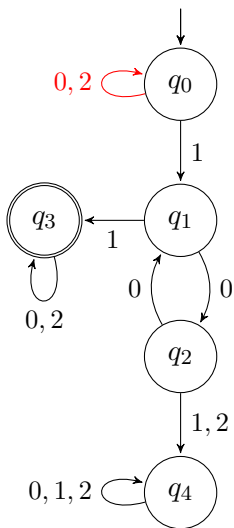
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

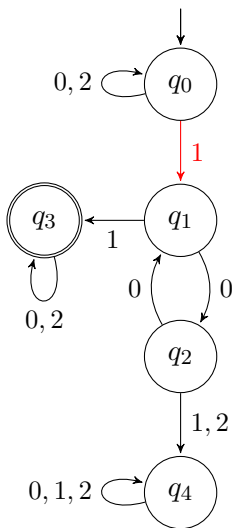
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

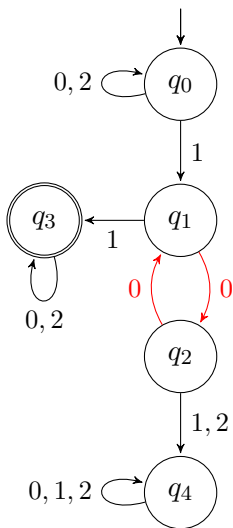
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^* \mathbf{1} (00)^* \mathbf{1} (0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

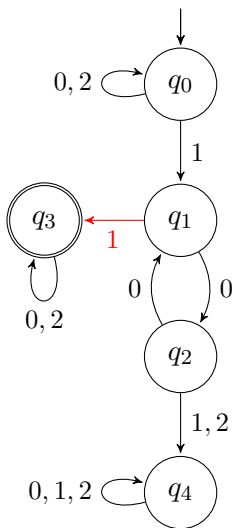
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^* 1 (00)^* 1 (0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

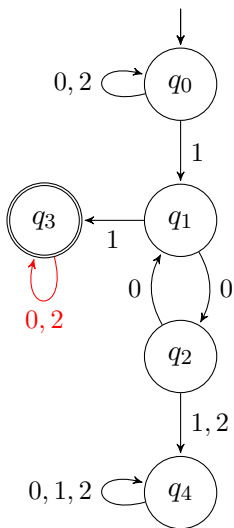
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

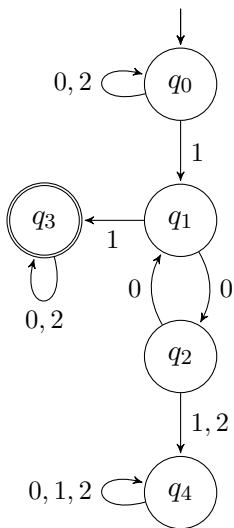
« Vérification » a posteriori



$$\mathcal{L}(A) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

« Vérification » a posteriori



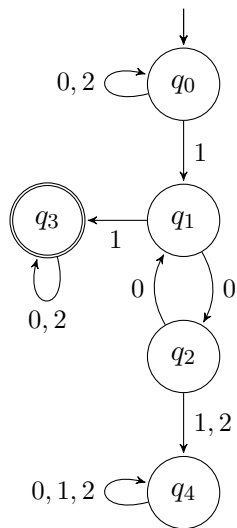
$$\mathcal{L}(A) = (0 + 2)^*1(00)^*1(0 + 2)^*$$

On peut s'assurer que l'automate « reconnaît » l'expression régulière calculée.

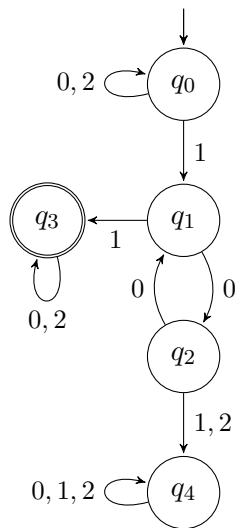
Remarque : on peut trouver différentes expressions régulières en fonction de l'ordre dans lequel les équations sont résolues.

On peut faire correspondre la substitution d'une variable et la suppression de l'état correspondant dans la méthode graphique.

Correspondance entre versions graphique et par équation



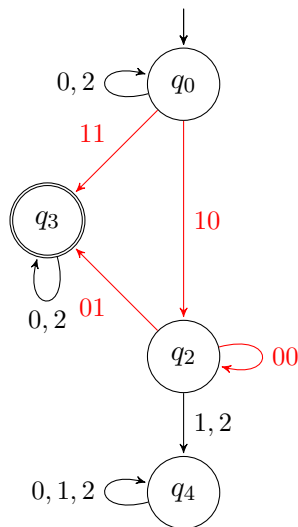
Correspondance entre versions graphique et par équation



Avant suppression :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

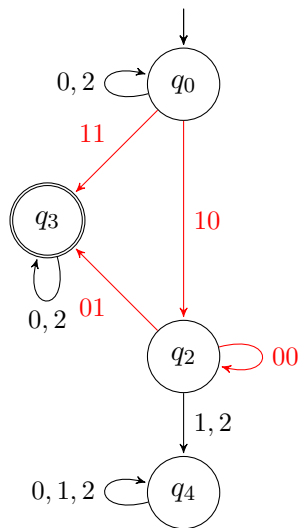
Correspondance entre versions graphique et par équation



Avant suppression :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Correspondance entre versions graphique et par équation



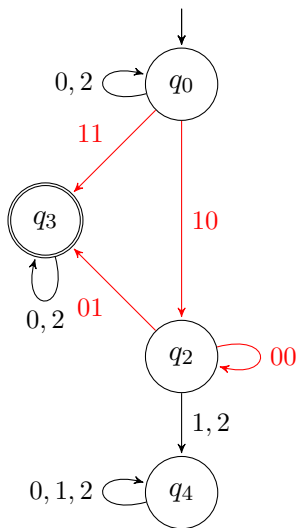
Avant suppression :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Après suppression de q_1 :

$$\begin{cases} x_0 = (0 + 2)x_0 + 10x_2 + 11x_3 \\ x_2 = 00x_2 + 01x_3 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Correspondance entre versions graphique et par équation



Avant suppression :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1x_1 \\ x_1 = 0x_2 + 1x_3 \\ x_2 = 0x_1 + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$

Après suppression de q_1 :

$$\begin{cases} x_0 = (0 + 2)x_0 + 1(0x_2 + 1x_3) \\ x_2 = 0(0x_2 + 1x_3) + (1 + 2)x_4 \\ x_3 = (0 + 2)x_3 + \epsilon \\ x_4 = (0 + 1 + 2)x_4 \end{cases}$$