

Langages et Automates

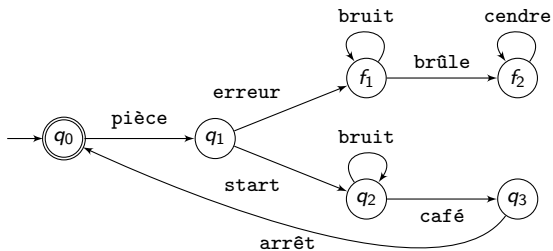
Les horreurs cosmiques

Engel Lefauchaux

Prépas des INP

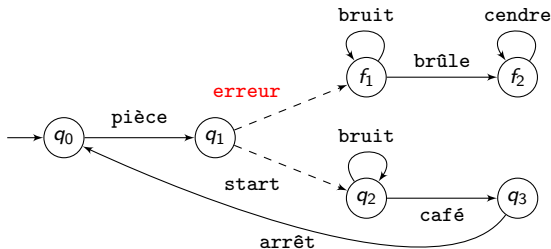
- Discuter de la modélisation de systèmes réels
 - Découvrir rapidement quelques formalismes plus poussés
- Discuter de la spécification de problèmes
 - Des mots, mais pas que

Un automate pour la machine à café



- les états peuvent correspondre aux situations du système
- les transitions peuvent correspondre aux événements / actions
 - Mais selon les problèmes, on voudra potentiellement ne pas représenter toutes les actions
- le choix d'un état final dépend de la propriété des mots que l'on veut tester.

Diagnostic de la machine à café



- les pointillés sont des actions inobservables par l'utilisateur
 - Action réelle du système, mais traitée comme une ε -transition
- Diagnostic : est-ce que tout mot correspondant à un chemin fautif est éventuellement détecté comme fautif ?
 - plus d'états acceptants
 - on considère même des mots infinis

Un monde probabiliste

Le hasard est omniprésent dans un système

- actions de l'environnement
- comportement intrinsèque du système.

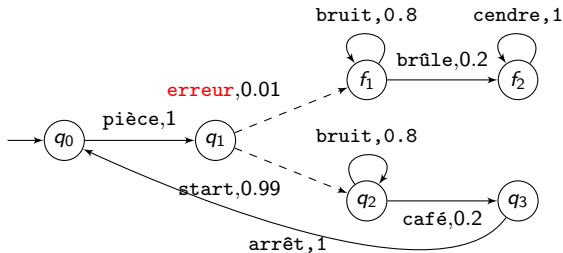
Comment le représenter ?

Un monde probabiliste

Le hasard est omniprésent dans un système

- actions de l'environnement
- comportement intrinsèque du système.

Comment le représenter ?



Première possibilité : chaque action a une probabilité

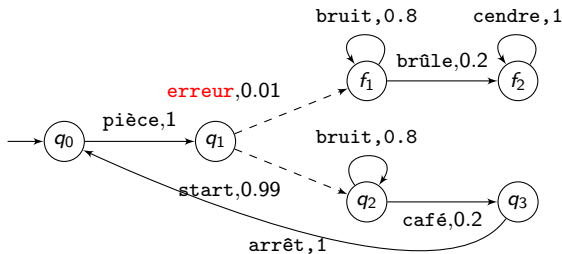
→ Chaîne de Markov étiquetée

Un monde probabiliste

Le hasard est omniprésent dans un système

- actions de l'environnement
- comportement intrinsèque du système.

Comment le représenter ?



Première possibilité : chaque action a une probabilité

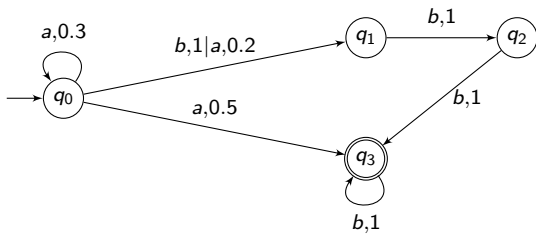
→ Chaîne de Markov étiquetée

Avec probabilité 1, une exécution contenant **erreur** contiendra également brûle.

Un monde probabiliste (2)

Deuxième possibilité : à chaque action est associé une distribution de probabilité indiquant l'état suivant

→ Automate probabiliste



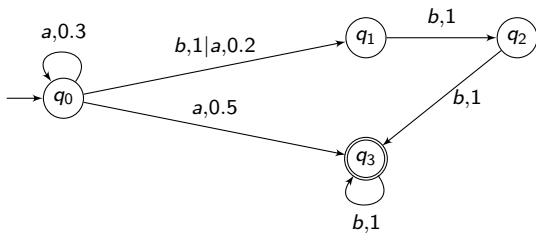
- lire b en q_0 mène en q_1
- lire a en q_0 mène en q_0 avec probabilité 0.3 , en q_1 avec probabilité 0.2 et en q_3 avec probabilité 0.5

Chaque mot a une probabilité de mener d'atteindre un état final.

Un monde probabiliste (2)

Deuxième possibilité : à chaque action est associé une distribution de probabilité indiquant l'état suivant

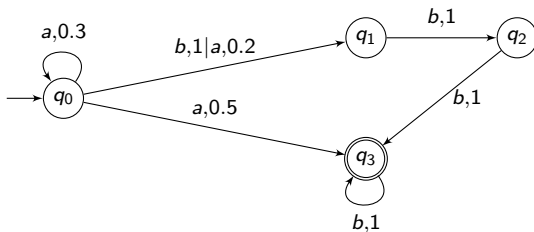
→ Automate probabiliste



- lire b en q_0 mène en q_1
- lire a en q_0 mène en q_0 avec probabilité 0.3, en q_1 avec probabilité 0.2 et en q_3 avec probabilité 0.5

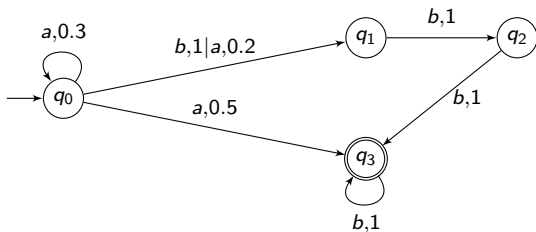
Chaque mot a une probabilité de mener d'atteindre un état final.
Quelle probabilité d'acceptation pour le mot $aabbbb$?

Langage probabiliste ?



Comment définir un langage pour un automate probabiliste ?

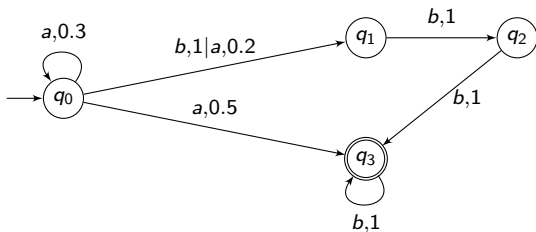
Langage probabiliste ?



Comment définir un langage pour un automate probabiliste ?

Traditionnellement, avec un seuil λ : le langage contient tous les mots dont la probabilité d'acceptation dépasse λ .

Langage probabiliste ?



Comment définir un langage pour un automate probabiliste ?

Traditionnellement, avec un seuil λ : le langage contient tous les mots dont la probabilité d'acceptation dépasse λ .

Quel est le langage de l'automate ci-dessus avec le seuil $\lambda = 0.5$?

Un monde temporisé

- Chaque évènement a un temps d'exécution
- Certaines questions sont fortement dépendantes du temps.

Comment le représenter ?

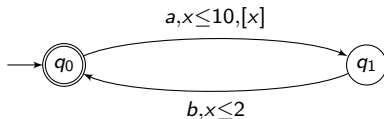
Une option : l'utilisation d'horloges

Un monde temporisé

- Chaque évènement a un temps d'exécution
- Certaines questions sont fortement dépendantes du temps.

Comment le représenter ?

Une option : l'utilisation d'horloges



- x est une horloge
- $[x]$ remet x à 0

$(a, 4)(b, 5)(a, 8)(b, 8.2)(a, 18)(b, 20)$ est-il accepté ?

Un monde infini

- un système fini (les automates de ce cours) conserve une quantité d'information limitée
- l'environnement envoie une quantité d'information non bornée

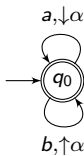
Comment retenir ces informations ?

Un monde infini

- un système fini (les automates de ce cours) conserve une quantité d'information limitée
- l'environnement envoie une quantité d'information non bornée

Comment retenir ces informations ?

Une possibilité : l'utilisation d'une pile



- $\downarrow \alpha$ ajoute α sur la pile
- $\uparrow \alpha$
 - ne peut se prendre que si α est au sommet de la pile
 - retire α du sommet de la pile.

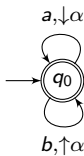
Quel langage pour cet automate ?

Un monde infini

- un système fini (les automates de ce cours) conserve une quantité d'information limitée
- l'environnement envoie une quantité d'information non bornée

Comment retenir ces informations ?

Une possibilité : l'utilisation d'une pile



- $\downarrow \alpha$ ajoute α sur la pile
- $\uparrow \alpha$
 - ne peut se prendre que si α est au sommet de la pile
 - retire α du sommet de la pile.

Quel langage pour cet automate ?

$$\{w \in \{a, b\}^* \mid \forall v, \text{ prefix de } w, |v|_a \geq |v|_b\}$$

Construisez un automate à pile reconnaissant les langages suivants

- $\{a^n b^n \mid n \in \mathbb{N}\}$
- $\{a^n b^m c^m d^n \mid n, m \in \mathbb{N}\}$

Le langage suivant peut-il être reconnu par un automate à pile :

$$\{a^n b^m c^n d^m \mid n, m \in \mathbb{N}\}$$

Langage hors contexte

Les langages reconnus par les automates à pile sont appelés langages hors-contexte.

Theorem

Théorème de la pompe (le Théorème de l'étoile des automates à pile) Soit L un langage hors contexte. Il existe $N \in \mathbb{N}$ tel que pour tout mot $w \in L$, si $|w| \geq N$, alors $w = uvwxy$ avec

- $|vwx| \leq N$
- $|vx| > 0$
- pour tout $k \in \mathbb{N}$, $uv^kwx^ky \in L$

Donc pour $\{a^n b^m c^n d^m \mid n, m \in \mathbb{N}\}$?

Langage hors contexte

Les langages reconnus par les automates à pile sont appelés langages hors-contexte.

Theorem

Théorème de la pompe (le Théorème de l'étoile des automates à pile) Soit L un langage hors contexte. Il existe $N \in \mathbb{N}$ tel que pour tout mot $w \in L$, si $|w| \geq N$, alors $w = uvwxy$ avec

- $|vwx| \leq N$
- $|vx| > 0$
- pour tout $k \in \mathbb{N}$, $uv^kwx^ky \in L$

Donc pour $\{a^n b^m c^n d^m \mid n, m \in \mathbb{N}\}$?

Les langages hors contexte correspondent aux langages générés par des grammaire de type 2.