Covers 3/3/21

$O(|V| + |E|)$

Algorithme linéaire pour calculer

$Attr_0(F)$ $\qquad F \subseteq V$

- File Q de sommets
- Tableaux R, done de sommets
- Tableau C (counter) :

$0 \leq C(v) \leq |Post(v)|$

On suppose que Pre, Post sont disponibles (sous forme de tableau de listes).

R : à la fin c'est $Attr_0(F)$

done : sommets déjà traités

Q : sommets en cours de traitement

C : utilisé juste pour V_1

Initialisation :

Pour tout $v \in V$ **faire**

    $C(v) := |Post(v)|$;

    $done(v) := false$

$Q \leftarrow \emptyset$; $R \leftarrow \emptyset$;

Pour tout $v \in F$ **faire**

    $Q \leftarrow enfiler(Q, v)$

    $done(v) := true$;

Tant que $(Q = \emptyset)$ **faire**

    $v \leftarrow defiler(Q)$

    $R := R \cup \{v\}$

? $v \leftarrow u$?

    Pour tout $(u \in Pre(v))$ **faire**

        si $(u \in V_0$ & $done(u) = false)$

        alors $Q \leftarrow enfiler(Q, u)$

            $done(u) := true$

        si $(u \in V_1$ & $done(u) = false)$

        alors $\boxed{C(u) := C(u) - 1}$;

            si $\boxed{C(u) = 0}$ **alors**

                $Q \leftarrow enfiler(Q, u)$

                $done(u) := true$

Retourner(R)

Propriétés

① À cause de "done", chaque sommet peut rentrer au plus 1 fois dans Q.

$\rightsquigarrow$ Enfiler(Q, u) est exécuté au plus 1 fois.

$\Rightarrow$ chaque arête ($u \in Pre(v)$) sera considéré au plus 1 fois

Tout autre opération : $O(1)$

$\Rightarrow$ complexité $O(|V| + |E|)$

② Pour tout sommet $u \in V$:

- si $u \in V_0$ et u est rajouté à R, alors $Post(u) \cap R = \emptyset$, ou $u \in F$.

init.

- si $u \in V_1$ et u est rajouté à R, alors :

$Post(u) \subseteq R$, ou $u \in F$

② $\Rightarrow$ $\quad R \subseteq \mathrm{Attr}_0(F)$

Stratégie d'attracteur pour P_0 :

$\sigma_0(u) = v$ si u a été rajouté à Q lors du traitement de v

③ Pour tout sommet $u \in V$ on a :

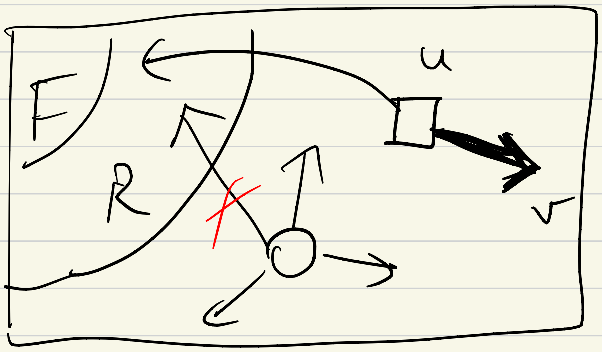
- si $u \in V_0$ et $\mathrm{Post}(u) \cap R \neq \emptyset$, alors l'algo rajoute u à R
- si $u \in V_1$ et $\mathrm{Post}(u) \subseteq R$, alors l'algo rajoute u à R.

$c(u) = 0$

$\rightsquigarrow$ u sera rajouté à Q, donc à R (plus tard)

Les $u \in V \setminus R$ ont la propriété :

- $u \in V_0$: $\quad \mathrm{Post}(u) \subseteq V \setminus R$
- $u \in V_1$: $\quad \mathrm{Post}(u) \cap (V \setminus R) \neq \emptyset$



$\sigma_1(u) = v$

$R = W_0$ région gagnante de P_0 (atteindre F)

$V \setminus R = W_1$ — " — de P_1 (éviter F)

Application (1) jeux pour la bisimulation

2 systèmes de transitions finis

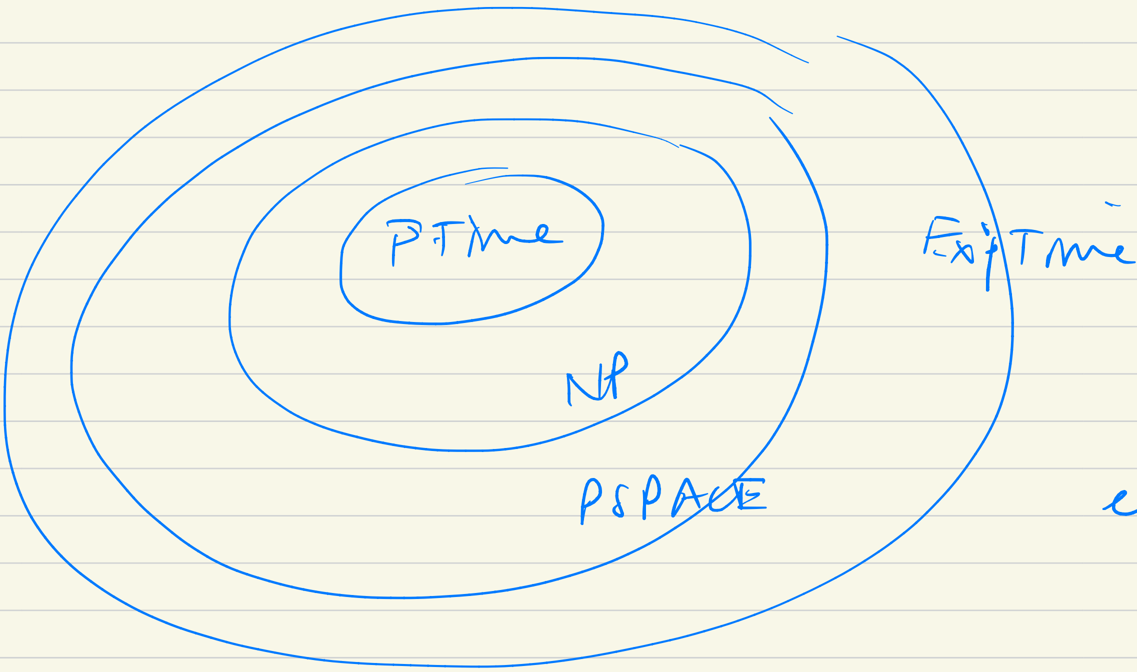
$A_i = (Q_i, \Sigma, q_{0,i}, \delta_i)$

$\delta_i \subseteq Q_i \times \Sigma \times Q_i$

[Avec états finaux on peut demander si $L(A_1) = L(A_2)$]

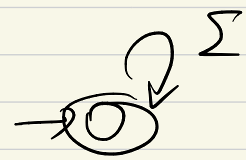
DFA (dét.) $\longrightarrow$ minimisation PTime

NFA (non-dét.) $\longrightarrow$ PSPACE-C.



PSPACE : espace poly

Pb. PSPACE-c :



Input NFA $\mathcal{A}$ | universalit

Q : $L(\mathcal{A}) = \Sigma^*$?

Input $A_1, \dots, A_n$ (N/D) FA

Q : $L(A_1) \cap \dots \cap L(A_n) \neq \emptyset$?

cas spécial de $L(A_1) \stackrel{?}{=} L(A_2)$

A_1, A_2 NFA

LTS $A_i = (Q_i, \Sigma, q_{0,i}, \delta_i)$

$$\delta_i \subseteq Q_i \times \Sigma \times Q_i$$

Une rel. de bisimulation R entre A_1, A_2 est une rel. binaire $R \subseteq Q_1 \times Q_2$

tq :

1. $(q_{0,1}, q_{0,2}) \in R$

2. $\forall (q_1, q_2) \in R \quad \forall a \in \Sigma$:

   2.1 $\forall\ q_1 \xrightarrow{a} q_1' \quad \exists\ q_2 \xrightarrow{a} q_2'$:
   $(q_1', q_2') \in R$

   2.2 $\forall\ q_2 \xrightarrow{a} q_2' \quad \exists\ q_1 \xrightarrow{a} q_1'$,
   $(q_1', q_2') \in R$

$$\begin{tikzcd}
q_1 \arrow[r, phantom, "R"] \arrow[d, "\forall\ a"'] & q_2 \arrow[d, dashed, "a\ \exists"] \\
q_1' \arrow[r, phantom, "R"] & q_2'
\end{tikzcd}
\qquad
\begin{tikzcd}
q_1 \arrow[r, phantom, "R"] \arrow[d, dashed, "\exists\ a"'] & q_2 \arrow[d, "a\ \forall"] \\
q_1' \arrow[r, phantom, "R"] & q_2'
\end{tikzcd}$$

3. Avec F_1, F_2 états finaux :
$$R \subseteq (F_1 \times F_2) \cup \left((Q_1 \setminus F_1) \times (Q_2 \setminus F_2) \right)$$

Jeu de bisimulation associé à A_1, A_2 :

– arène : $\underbrace{(Q_1 \times Q_2)}_{= V_0} \cup \underbrace{(Q_1 \times Q_2 \times \Sigma \times \{1,2\})}_{= V_1}$

coup :

$(q_1, q_2) \longrightarrow (q_1', q_2', a, i)$ si
$\in V_0$

ou $\begin{cases} i=1, & q_1 \xrightarrow[1]{a} q_1', \quad q_2' = q_2 \\ i=2, & q_2 \xrightarrow[2]{a} q_2', \quad q_1' = q_1 \end{cases}$

$(q_1, q_2, a, i) \rightarrow (q_1', q_2')$ si

ou $\begin{cases} i=1, & q_2 \xrightarrow[2]{a} q_2', \quad q_1' = q_1 \\ i=2, & q_1 \xrightarrow[1]{a} q_1', \quad q_2' = q_2 \end{cases}$

P_0 : Spoiler (veut montrer qu'il n'y a pas de bisim.)

P_1 : Duplicator (veut montrer qu'il y a une bisim.)

P_0 gagne si on arrive dans

$$\boxed{\begin{array}{lll} & (q_1, q_2, a, i) \text{ tq.} & \\ i=1 & \text{et} \quad q_2 \overset{a}{\not\to}_2 & , \text{ou} \\ i=2 & \text{et} \quad q_1 \overset{a}{\not\to}_1 & \end{array}}$$

F

P_1 gagne toute partie infinie, ainsi que les parties qui finissent dans états finis (q_1, q_2) :

$$\forall a \in \Sigma \qquad q_1 \overset{a}{\not\to}_1 \ , \ q_2 \overset{a}{\not\to}_2$$

<u>Propriété</u> :

$(q_{0,1}, q_{0,2})$ gagnant pour P_0 $\Leftrightarrow$ il n'y a pas de relation de bisimulation entre A_1, A_2

$R = \{(q_1, q_2) : (q_1, q_2)$ gagnant pour P_1 dans le jeu de bisimulation$\}$ est bisimulation

Application ②

Automates finis
- dét. → unique calcul
- non-dét. → $\exists$
- alternants

$\mathcal{A}$ NFA, $w \in \Sigma^*$

$w \in L(\mathcal{A})$ s'il ex. calcul acc. de $\mathcal{A}$ sur w

Dual: $w \in L(\mathcal{A})$ si tous les calculs de $\mathcal{A}$ sur w sont acceptants

Un automate alternant

$\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$

Q états, $F \subseteq Q$, $q_0 \in Q$

$\delta : Q \times \Sigma \to \mathrm{Bool}_+(Q)$

formules bool. sans nég. sur Q

ex

$\varphi = (q_1 \wedge q_2) \vee q_3 \quad \in \mathrm{Bool}_+(Q)$

$p \xrightarrow{a} (q_1 \wedge q_2) \vee q_3$

choix universel

choix non-dét.

$R \subseteq Q$, $\varphi \in \mathrm{Bool}_+(Q)$

On écrit $\boxed{R \models \varphi}$ si R satisfait φ, c-a-d

si φ est vraie pour la valuation

$$val(q) = \begin{cases} true & \text{si } q \in R \\ false & \text{sinon} \end{cases}$$

$\{q_1, q_2\} \models \varphi$, $\{q_3\} \models \varphi$

$\{q_1, q_2, q_3\} \models \varphi$

$$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$$

$$w = a_1 \dots a_n \in \Sigma^*$$

Jeu $G(\mathcal{A}, w)$:

- arène $\underbrace{(Q \times \{1, \dots, n\})}_{=: V_0} \cup \underbrace{(2^Q \times \{1, \dots, n\})}_{=: V_1}$

- cmpl :

$$(q, i) \longrightarrow (R, i) \qquad \text{si } R \subseteq Q$$

est tq. $R \models \delta(q, a_i)$

$$(R, i) \rightarrow (q', i+1) \quad \text{tq.}$$

$$q' \in R$$

On suppose que $\forall q \in Q, \forall a \in \Sigma$:

$$\delta(q, a) \neq \text{false}, \text{true}$$

Condition de victoire pour P_0 dans $G(A, w)$

$\rightarrow$ atteindre $F \times \{n\}$

$$\boxed{w \in L(A) \stackrel{def}{\iff} P_0 \text{ gagne ce jeu d'accessibilité à partir de } (q_0, 0)}$$

Ex. $L = \{ ww : w \in \{a,b\}^n \}$

Tout NFA A tq. $L(A) = L$ est de taille exp. (A doit mémoriser la première moitié).

Mais il ex. un aut. alt. B de taille poly tq. $L(B) = L$

B : pour chaque $1 \leq i \leq n$:

la $i$ème lettre $=$ la $(n+i)$ème lettre

$\rightsquigarrow Q = \{1, \dots, n\}^2 \times \Sigma \cup \{q_0\} \cup \{1, \dots, n\}$

$$\delta\left(q_0, \substack{a \\ b}\right) = 1 \wedge \left(1, 0, \substack{a \\ b}\right)$$

$$\delta\left(1, \substack{a \\ b}\right) = 2 \wedge \left(2, 0, \substack{a \\ b}\right)$$

$$\vdots$$

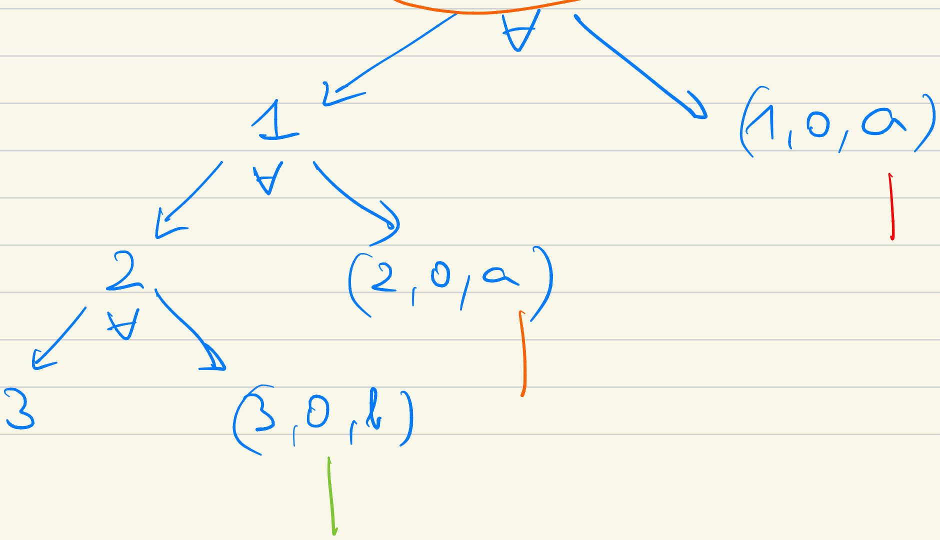
$$\delta\left(n, \substack{a \\ b}\right) = n \wedge \left(n, 0, \substack{a \\ b}\right) \qquad \underbrace{1 \, \cdots \, n}\ \underbrace{1 \, \cdots \, n}$$

$$\delta((i,0,a),c) = (i,1,a) \qquad , c \in \{a,b\}$$

$$\delta((i,j,a),c) = \begin{cases} (i,j+1,a) & \text{si } j \neq n \\ \text{true} & \text{si } j = n, c = a \end{cases}$$

q_0

a a b a a b $\qquad n = 3$



$$\begin{tikzcd}
\mathrm{DFA} \arrow[d] \arrow[r, phantom, "="] & \mathrm{NFA} \arrow[d] \arrow[l, bend right, "\exp"'] \arrow[r, phantom, "="] & \mathrm{AFA} \arrow[d] \arrow[l, bend right, "\exp."'] \\
\text{dét.} & \text{non-dét.} & \text{alternant}
\end{tikzcd}$$

aut. finis

$A_1, A_2, \dots, A_n$ NFA de taille $\leq m$

$A_i = (Q_i, \Sigma, q_{0,i}, \delta_i, F_i)$

AFA A tq.

$$L(A) = \bigcap_{i=1}^{n} L(A_i)$$

de taille $O(n \cdot m)$

$$Q = \bigcup_{i=1}^{n} Q_i \times \{1, \dots, n\} \cup \{q_0\}$$

$$\delta(q_0, a) = \bigwedge_{i=1}^{n} \bigvee_{q_{0,i} \xrightarrow{a}_i q_i} (q_i, i)$$

intersection des A_i

$$\delta((q,i), a) = \bigvee_{q \xrightarrow{a}_i p} (p, i)$$

$$F = \bigcup F_i \times \{i\}$$