

$P = \text{langages décidables en temps poly par une MT}$
 $NP = \text{" " " " " " " " par une MT non-déterministe}$

$P = NP ?$

• Temps d'une MT: # de transitions faites avant l'arrêt

• Une MT M est en temps $O(f(n))$ si:
 Entrée w , M prend un temps $O(f(n))$, où $|w|=n$

• Espace d'une MT: # de cellules distinctes utilisées



Espace \leq Temps

$DTIME(f(n))$: un langage L est dans $DTIME(f(n))$
 s'il ∃ un MT qui décide L
 en temps $O(f(n))$

Classe de langages P (poly)

$$P = \bigcup_{k \in \mathbb{N}} DTIME(n^k)$$

= plus résolvables en temps "raisonnable" (n^{100} raisonnable)

$$\text{ex: } L_+ = \{ \langle a, b, c \rangle : a + b = c \} \in P$$

$PATH = \{ \langle G, s, t, k \rangle : G \text{ est un graphe dans lequel}$
 le chemin le + court de
 $s \rightarrow t \text{ est } \leq k \}$

fouille en largeur $O(|V(G)| + |E(G)|)$ |
 sommets arêtes
 Dijkstra $O(|V(G)| \log |E(G)|)$

MT non-déterministe

Q : états

Σ : alphabet

$q_0 \in Q$: état initial

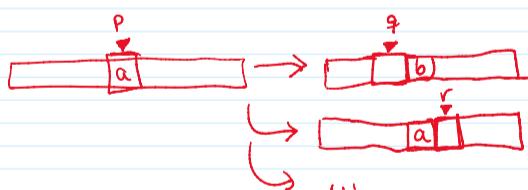
(blank) w : vide

$A \subseteq Q$: états acceptants

MT dét. $\delta: Q \setminus A \times \Sigma \rightarrow Q \times \Sigma \times \{L, R\}$

$$\delta \subseteq (Q \setminus A \times \Sigma) \times (Q \times \Sigma \times \{L, R\})$$

état-symbole état-symbole-gauche/droite



Intuition

MT $w \rightarrow q_{f_0} \rightarrow q_{f_1} \rightarrow q_{f_2} \rightarrow \dots$

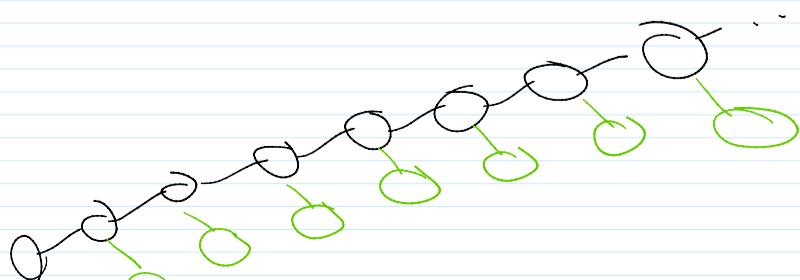
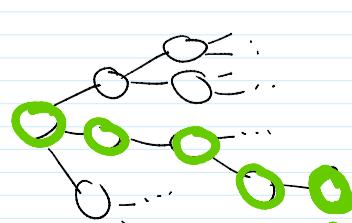
MT non-dét. $w \rightarrow q_{f_0} \xrightarrow{\text{multiple paths}} \dots$

Acceptation

Une MT non-det. M accepte $w \in \Sigma^*$
 s'il existe une séquence de choix de transitions
 qui mènent à un état acceptant.
 Sinon, M rejette w.

Temps d'une MT non-dét

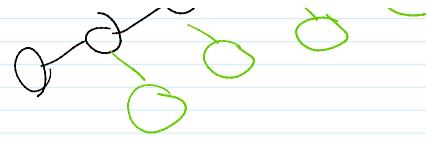
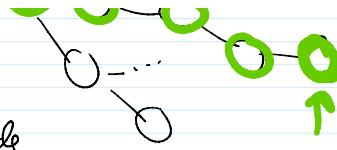
Sur entrée w , le temps de M:



Temps < une entrée

Sur entrée w , le temps de M :

- si w est accepté, le temps de M est le # de transitions sur le chemin le + court menant à l'acceptation de M .
- si w n'est pas accepté, le temps de M est 1.



$\text{NTIME}(f(n))$: un langage L est dans $\text{NTIME}(f(n))$

s'il existe une MT non-déterm. dont le langage accepté est L et dont le temps est tjs $O(f(n))$.

$$\text{NP} \subseteq \bigcup_{k \in \mathbb{N}} \text{NTIME}(n^k)$$

ex: $SAT = \{\varphi : \varphi \text{ est une formule booléenne satisfaisable}\}$

$$\varphi: (x_1 \vee x_2 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee x_4 \vee \bar{x}_1) \wedge (\bar{x}_1 \vee \bar{x}_4)$$

$$\underbrace{F \ F \ V}_{F} \wedge ? \wedge ? \dots$$

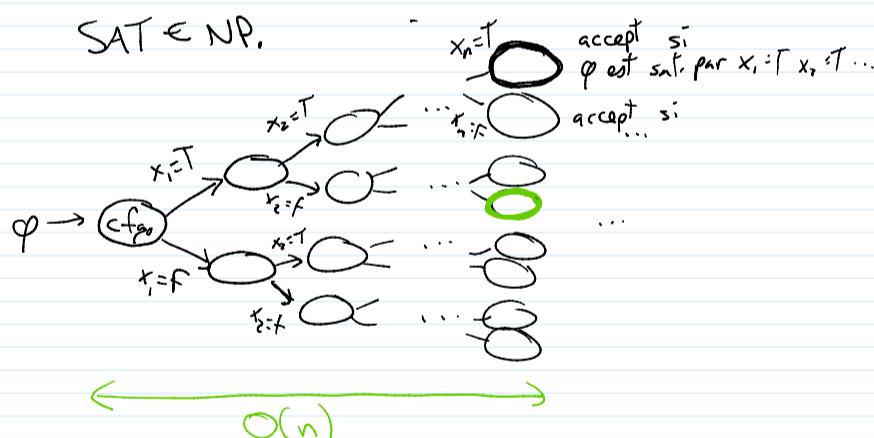
$\varphi \in SAT$ si on peut assigner $x_i = T$ ou $x_i = F$ à x_i de façon à ce que φ évalue à T .

$$\begin{array}{ll}
 x_1 = V & (x_1 \vee x_2 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee x_4 \vee \bar{x}_1) \wedge (\bar{x}_1 \vee \bar{x}_4) \\
 x_2 = F & (V \vee F \vee V) \wedge (V \vee F \vee F) \wedge (F \vee V) \\
 x_3 = F & V \wedge V \wedge V \\
 x_4 = F & V \in SAT
 \end{array}$$

$x_1 \wedge \bar{x}_1 \notin SAT$

On ne sait pas si $SAT \in P$.

$SAT \in NP$.



Déf. équivalente de NP par vérificateur

Soit L un langage. Une MT V est un vérificateur polynomial pour L si:

- $w \in L \Leftrightarrow \exists c \in \Sigma^*$ tel que V accepte $\langle w, c \rangle$ et $|c| \in O(|w|^k)$, $k \in \mathbb{N}$
- V s'exécute en temps polynomial

$$w \in L: \quad \boxed{\exists c} \quad \boxed{V} \rightarrow w \in L \quad \text{accept}$$

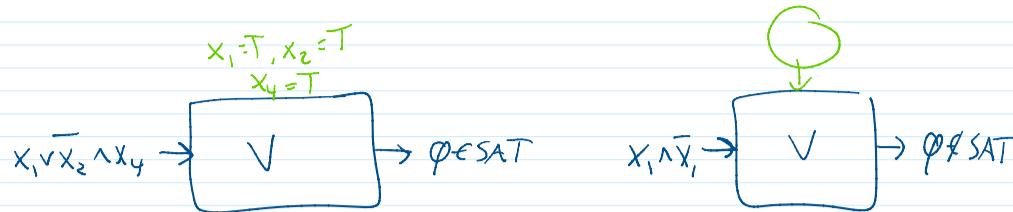
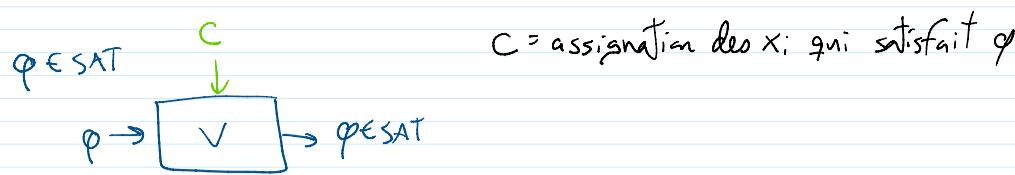
$$w \notin L: \quad \boxed{w \notin L} \quad \boxed{V} \rightarrow \boxed{w \notin L} \quad \text{rejet}$$

ex: vérif. pour SAT

$$\varphi \in SAT$$

$$\boxed{C}$$

$C = \text{assignation des } x_i \text{ qui satisfait } \varphi$



Intuition: c est une "preuve" que le mot est dans le langage.

Si le mot n'est pas dans le langage, toute preuve échouera.

Vérificateur pour SAT

sur entrée $\langle \varphi, c \rangle$

verifier que c encode une assignation de x_i :

(sinon, rejeter)

si c satisfait φ

accepter

sinon

rejeter

Thm: $L \in \text{NP} \iff \exists \text{ un vérificateur polynomial pour } L$

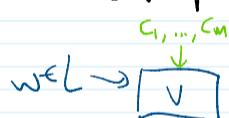
$(\Rightarrow) L \in \text{NP} \Rightarrow \exists \text{ vérif.}$

Soit $L \in \text{NP}$. Alors, \exists MT non-déf. M dont langage est L et qui prend un temps $O(n^k)$, $k \in O(1)$.

On utilise M pour construire un vérif. V .

[Soit $w \in L$. Soit c_1, \dots, c_m la séquence de configs de M menant à accepter w .

Le vérif. V attend (c_1, \dots, c_m) comme certificat que $w \in L$.



V : sur entrée $\langle w, (c_1, \dots, c_m) \rangle$

Soit M la MT non-déf. dont le langage est L

Verifier que c_1 est la cfg initiale de M sur w

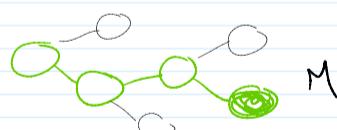
pour $i=1, \dots, m-1$
| si c_i ne peut pas mener à c_{i+1} sur M
| | rejeter

| si c_m est acceptant

| | accepter

sinon

| rejeter



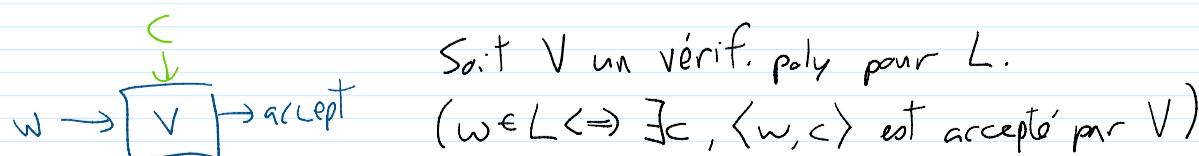
M

Si $w \in L$, V acceptera w si on lui passe c_1, \dots, c_m empruntés par M , puisque M accepte w à c_m .

Si $w \notin L$, V rejettéra w car M n'a aucune seq. de transitions acceptant w .

ex: V est en temps polynomial

$(\Leftarrow) \exists \text{ vérif. } V \text{ pour } L \Rightarrow L \in \text{NP}$



On construit une MT non-déf pour L .

Soit M qui, sur entrée w :

" . - + " "+ - . " " ∨ - " " ∩ - " " ∪ - "

Soit M qui, sur entrée w :

- "choisit" un certificat c de façon de façon non-déterm.
- simule V sur $\langle w, c \rangle$
- accepte si V a accepté