COMPLEX - Cours 10
Classes de complexité probabilistes

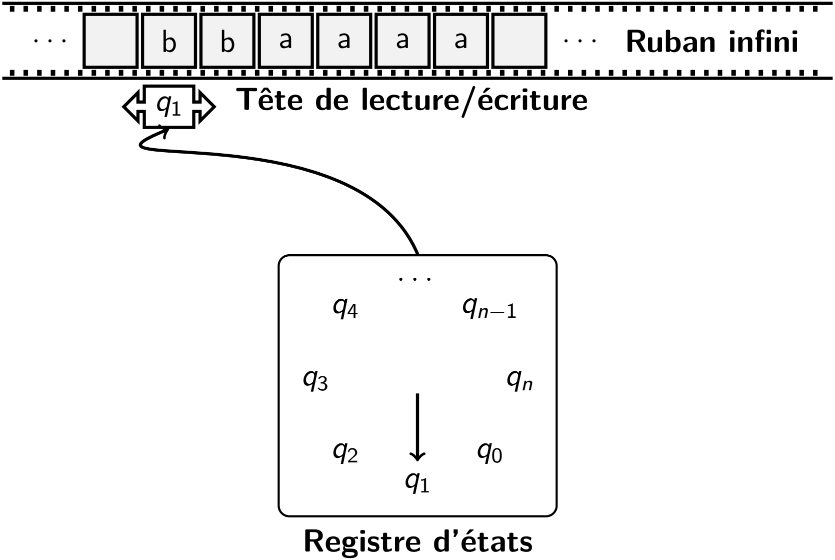
Damien Vergnaud

Sorbonne Université – CNRS

Table des matières

1. Définitions
   - Rappels
   - Machines de Turing probabilistes
2. Classe de complexité $\mathcal{BPP}$
   - Définition
   - Réduction de l'erreur et classe $\mathcal{PP}$
3. Classes de complexité $\mathcal{RP}$, $co-\mathcal{RP}$ et $\mathcal{ZPP}$
   - Classes de complexité $\mathcal{RP}$ et $co-\mathcal{RP}$
   - Classe de complexité $\mathcal{ZPP}$
4. Conclusion

Machines de Turing – Rappel

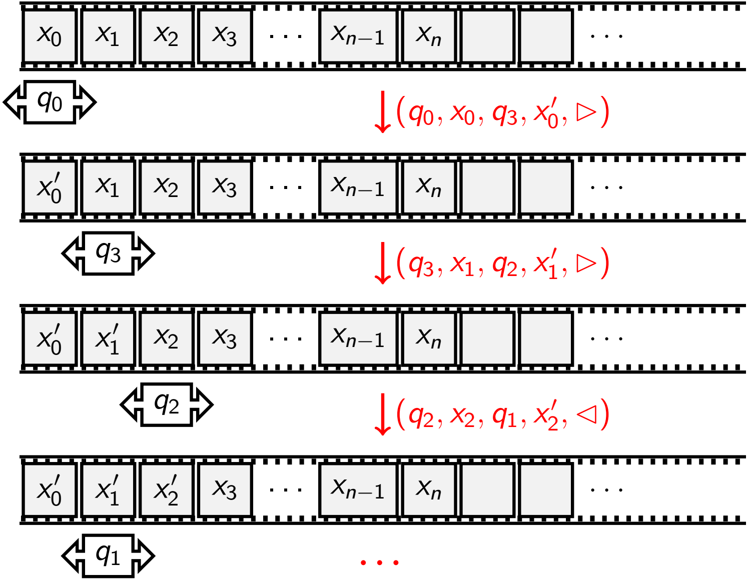


Machines de Turing – Rappel

Une **machine de Turing** est un quintuplet $(Q, \Gamma, q_0, q_n, \delta)$ où :

- $Q = \{q_0, \ldots, q_n\}$ est un ensemble fini d'*états*
- Γ est l'*alphabet de travail* des symboles de la bande
  avec $\square$ un symbole particulier (dit *blanc*), $\square \in \Gamma$
- q_0 est l'état *initial*
- q_n est l'état *acceptant*
- $\delta : Q \times \Gamma \to Q \times \Gamma \times \{\triangleleft, \triangleright\}$ est la fonction de *transition*

Machines de Turing – Rappel



Machines de Turing – Rappel

- Nous supposons que $\mathcal{M}$ s'arrête sur tout $x \in \Sigma^*$ (avec $\Sigma \subset \Gamma$)
  - $\mathcal{M}$ arrive dans une configuration avec l'état q_n
    $\rightsquigarrow \mathcal{M}(x) = 1$
  - $\mathcal{M}$ arrive dans une configuration sans transition possible
    $\rightsquigarrow \mathcal{M}(x) = 0$

La classe $\mathcal{DTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \rightarrow \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{DTIME}(T)$ si et seulement si il existe une machine de Turing $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur $x \in \Sigma^*$ en temps au plus $T(|x|)$
- Pour tout $x \in \mathcal{L}$, nous avons $\mathcal{M}(x) = 1$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\mathcal{M}(x) = 0$

Machines de Turing – Rappel

- Nous supposons que $\mathcal{M}$ s'arrête sur tout $x \in \Sigma^*$ (avec $\Sigma \subset \Gamma$)
  - $\mathcal{M}$ arrive dans une configuration avec l'état q_n
    $\rightsquigarrow \mathcal{M}(x) = 1$
  - $\mathcal{M}$ arrive dans une configuration sans transition possible
    $\rightsquigarrow \mathcal{M}(x) = 0$

La classe $\mathcal{P}$

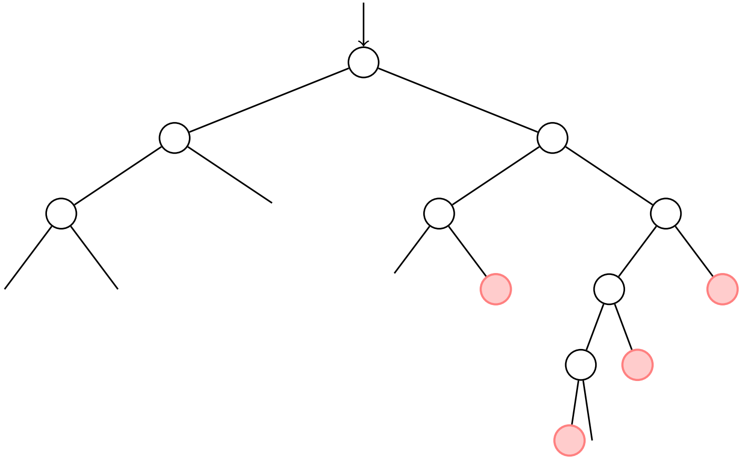
$$\mathcal{P} = \bigcup_{k \in \mathbb{N}} \mathcal{DTIME}(n \longmapsto n^k)$$

Machines de Turing non-déterministes – Rappel

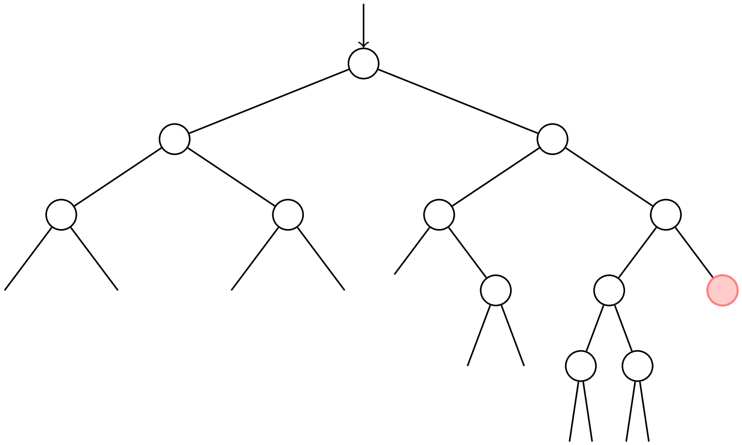
Une **machine de Turing non-déterministe** est un sextuplet $(Q, \Gamma, q_0, q_n, \delta_0, \delta_1)$ où :

- $Q = \{q_0, \ldots, q_n\}$ est un ensemble fini d'*états*
- Γ est l'*alphabet de travail* des symboles de la bande avec $\square$ un symbole particulier (dit *blanc*), $\square \in \Gamma$
- q_0 est l'état *initial*
- q_n est l'état *acceptant*
- $\delta_0 : Q \times \Gamma \to Q \times \Gamma \times \{\lhd, \rhd\}$ et $\delta_1 : Q \times \Gamma \to Q \times \Gamma \times \{\lhd, \rhd\}$ sont les deux fonctions de *transition*

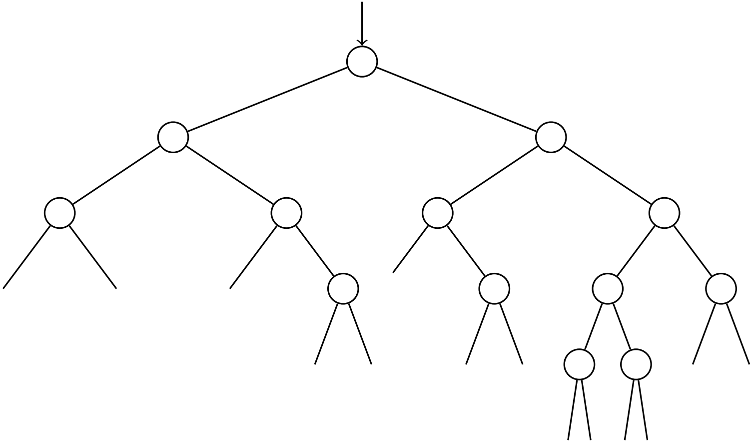
Machines de Turing non-déterministes – Rappel



Machines de Turing non-déterministes – Rappel



Machines de Turing non-déterministes – Rappel



Machines de Turing non-déterministes – Rappel

- Nous supposons que $\mathcal{M}$ s'arrête sur tout $x \in \Sigma^*$ (avec $\Sigma \subset \Gamma$) (pour tous les choix de fonctions de transition)
  - $\mathcal{M}$ arrive ≥ 1 fois dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 1$
  - $\mathcal{M}$ n'arrive jamais dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 0$

La classe $\mathcal{NTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \to \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{NTIME}(T)$ si et seulement si il existe une machine de Turing $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur $x \in \Sigma^*$ en temps au plus $T(|x|)$
- Pour tout $x \in \mathcal{L}$, nous avons $\mathcal{M}(x) = 1$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\mathcal{M}(x) = 0$

Machines de Turing non-déterministes – Rappel

- Nous supposons que $\mathcal{M}$ s'arrête sur tout $x \in \Sigma^*$ (avec $\Sigma \subset \Gamma$) (pour tous les choix de fonctions de transition)
  - $\mathcal{M}$ arrive ≥ 1 fois dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 1$
  - $\mathcal{M}$ n'arrive jamais dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 0$

La classe $\mathcal{NP}$

$$\mathcal{NP} = \bigcup_{k \in \mathbb{N}} \mathcal{NTIME}(n \longmapsto n^k)$$

Machines de Turing non-déterministes – Rappel

- Nous supposons que $\mathcal{M}$ s'arrête sur tout $x \in \Sigma^*$ (avec $\Sigma \subset \Gamma$) (pour tous les choix de fonctions de transition)
  - $\mathcal{M}$ arrive ≥ 1 fois dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 1$
  - $\mathcal{M}$ n'arrive jamais dans une configuration avec l'état q_n $\rightsquigarrow \mathcal{M}(x) = 0$

La classe $\mathcal{NP}$

$$\mathcal{NP} = \bigcup_{k \in \mathbb{N}} \mathcal{NTIME}(n \longmapsto n^k)$$

La classe $\mathrm{co} - \mathcal{NP}$

$$\mathcal{L} \in \mathrm{co} - \mathcal{NP} \iff \Sigma^* \setminus \mathcal{L} \in \mathcal{NP}$$

Machines de Turing non-déterministes – Rappel

Une **machine de Turing non-déterministe** est un sextuplet $(Q, \Gamma, q_0, q_n, \delta_0, \delta_1)$ où :

- $Q = \{q_0, \ldots, q_n\}$ est un ensemble fini d'*états*
- Γ est l'*alphabet de travail* des symboles de la bande avec $\Box$ un symbole particulier (dit *blanc*), $\Box \in \Gamma$
- q_0 est l'état *initial*
- q_n est l'état *acceptant*
- $\delta_0 : Q \times \Gamma \to Q \times \Gamma \times \{\lhd, \rhd\}$ et $\delta_1 : Q \times \Gamma \to Q \times \Gamma \times \{\lhd, \rhd\}$ sont les deux fonctions de *transition*

Machines de Turing probabilistes

Une **machine de Turing probabiliste** est un sextuplet $(Q, \Gamma, q_0, q_n, \delta_0, \delta_1)$ où :

- $Q = \{q_0, \ldots, q_n\}$ est un ensemble fini d'*états*
- Γ est l'*alphabet de travail* des symboles de la bande avec $\square$ un symbole particulier (dit *blanc*), $\square \in \Gamma$
- q_0 est l'état *initial*
- q_n est l'état *acceptant*
- $\delta_0 : Q \times \Gamma \to Q \times \Gamma \times \{\triangleleft, \triangleright\}$ et $\delta_1 : Q \times \Gamma \to Q \times \Gamma \times \{\triangleleft, \triangleright\}$ sont les deux fonctions de *transition*

Machines de Turing probabilistes

- La définition syntaxique est identique à celle des machines non-déterministes

- La différence vient de l'interprétation du graphe des exécutions de la machine
  - Nous ne regardons plus si il existe un chemin menant à l'état acceptant
  - Nous regardons la proportion de ces chemins

- Plus précisément, nous interprétons le graphe comme :

À chaque étape de calcul $\mathcal{M}$ exécute l'une des deux fonctions de transition δ_0 et δ_1 (tiré uniformément aléatoirement)

Machines de Turing probabilistes

- La définition syntaxique est identique à celle des machines non-déterministes
- La différence vient de l'interprétation du graphe des exécutions de la machine
  - Nous ne regardons plus si il existe un chemin menant à l'état acceptant
  - Nous regardons la proportion de ces chemins
- Plus précisément, nous interprétons le graphe comme :

À chaque étape de calcul $\mathcal{M}$ exécute l'une des deux fonctions de transition δ_0 et δ_1 (tiré uniformément aléatoirement)

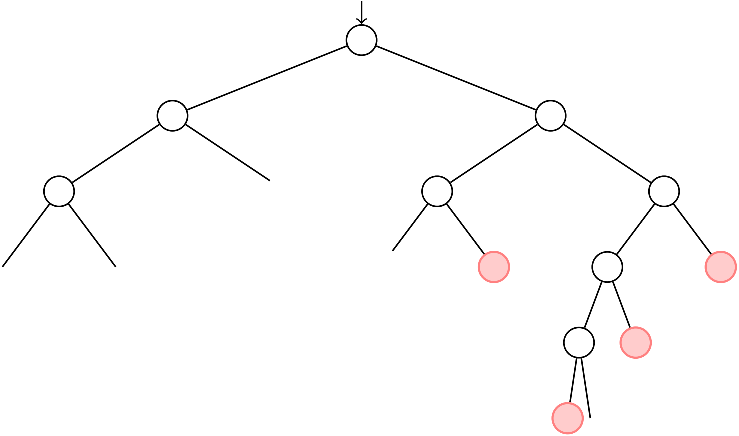
Machines de Turing probabilistes

- La définition syntaxique est identique à celle des machines non-déterministes
- La différence vient de l'interprétation du graphe des exécutions de la machine
  - Nous ne regardons plus si il existe un chemin menant à l'état acceptant
  - Nous regardons la proportion de ces chemins
- Plus précisément, nous interprétons le graphe comme :

À chaque étape de calcul $\mathcal{M}$ exécute l'une des deux fonctions de transition δ_0 et δ_1 (tiré uniformément aléatoirement)

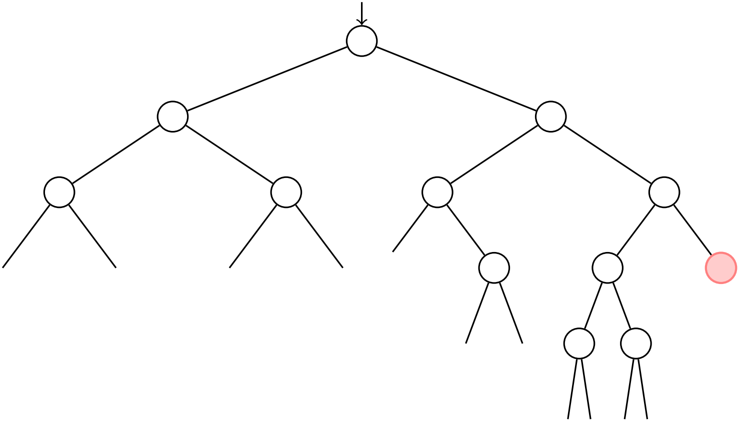
Machines de Turing probabilistes

$$\Pr(\mathcal{M} \text{ accepte}) = \frac{1}{8} + \frac{1}{32} + \frac{1}{16} + \frac{1}{8} = \frac{11}{32}$$



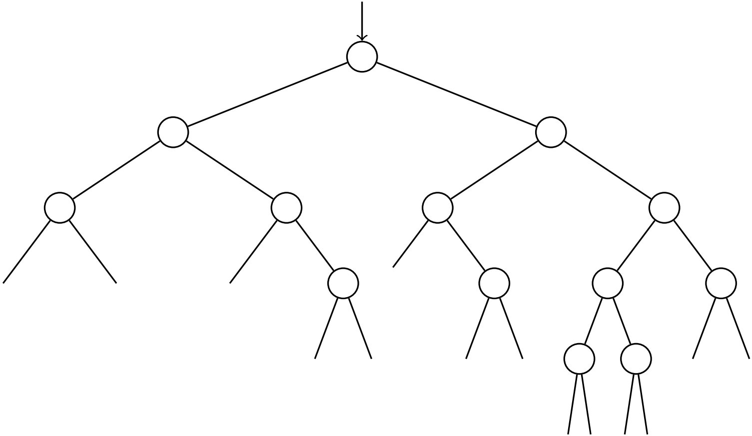
Machines de Turing probabilistes

$$\Pr(\mathcal{M} \text{ accepte}) = \frac{1}{8}$$



Machines de Turing probabilistes

$\Pr(\mathcal{M} \text{ accepte}) = 0$



Définition de l'acceptation

- Nous ne pouvons plus dire qu'une machine de Turing probabiliste $\mathcal{M}$ accepte ou rejette un mot
- Nous définissons donc une variable aléatoire

$$\mathcal{M}(x) = \begin{cases} 1 & \text{si } \mathcal{M} \text{ s'arrête en son état acceptant sur } x \\ 0 & \text{sinon} \end{cases}$$

sur l'expérience aléatoire consistant à exécuter $\mathcal{M}$ en tirant uniformément aléatoirement une des deux fonctions de transition à chaque étape de calcul.

Définition de l'acceptation

- Nous ne pouvons plus dire qu'une machine de Turing probabiliste $\mathcal{M}$ accepte ou rejette un mot
- Nous définissons donc une variable aléatoire

$$\mathcal{M}(x) = \begin{cases} 1 & \text{si } \mathcal{M} \text{ s'arrête en son état acceptant sur } x \\ 0 & \text{sinon} \end{cases}$$

sur l'expérience aléatoire consistant à exécuter $\mathcal{M}$ en tirant uniformément aléatoirement une des deux fonctions de transition à chaque étape de calcul.

Table des matières

1 Définitions
- Rappels
- Machines de Turing probabilistes

2 Classe de complexité $\mathcal{BPP}$
- Définition
- Réduction de l'erreur et classe $\mathcal{PP}$

3 Classes de complexité $\mathcal{RP}$, $co - \mathcal{RP}$ et $\mathcal{ZPP}$
- Classes de complexité $\mathcal{RP}$ et $co - \mathcal{RP}$
- Classe de complexité $\mathcal{ZPP}$

4 Conclusion

La classe $\mathcal{BPP}$

La classe $\mathcal{BPTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \to \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPTIME}(T)$ si et seulement si il existe une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $T(|x|)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{2}{3}$

La classe $\mathcal{BPP}$

$$\mathcal{BPP} = \bigcup_{k \in \mathbb{N}} \mathcal{BPTIME}(n \longmapsto n^k)$$

La classe $\mathcal{BPP}$

La classe $\mathcal{BPTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \to \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPTIME}(T)$ si et seulement si il existe une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $T(|x|)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{2}{3}$

La classe $\mathcal{BPP}$

$$\mathcal{BPP} = \bigcup_{k \in \mathbb{N}} \mathcal{BPTIME}(n \longmapsto n^k)$$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{2}{3}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

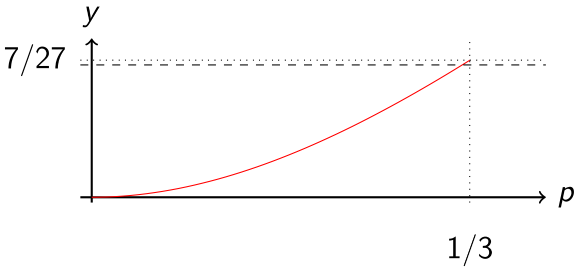
- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{3}{4}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{3}{4}$

Réduction de l'erreur

Vote majoritaire 2 (ou plus) parmi 3

Exécution 1	1	1	1	1	0	0	0	0
Exécution 2	1	1	0	0	1	1	0	0
Exécution 3	1	0	1	0	1	0	1	0
Résultat	1	1	1	0	1	0	0	0

$$f(p) = p^3 + 3p^2 \cdot (1-p) \text{ et } \max f(p) = f\left(\frac{1}{3}\right) = \frac{7}{27} > \frac{1}{4}$$

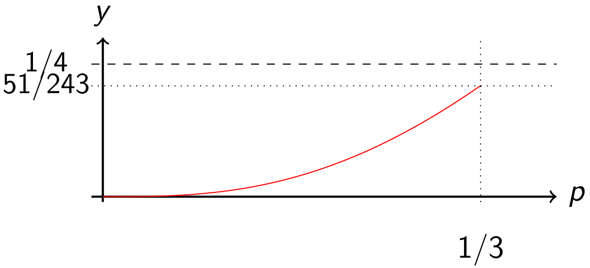


Réduction de l'erreur

Vote majoritaire 3 (ou plus) parmi 5

$$g(p) = p^5 + 5 \cdot p^4 \cdot (1-p) + 10 \cdot p^3 \cdot (1-p)^2$$

$$\max g(p) = g\left(\frac{1}{3}\right) = \frac{51}{243} < \frac{1}{4}$$



Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{2}{3}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{3}{4}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{3}{4}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{4}{5}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{4}{5}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.

Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \dfrac{1}{2} + \varepsilon$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \dfrac{1}{2} + \varepsilon$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.

Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq 1 - \dfrac{1}{2^{|x|^c}}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq 1 - \dfrac{1}{2^{|x|^c}}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \dfrac{1}{2} + \dfrac{1}{|x|^c}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \dfrac{1}{2} + \dfrac{1}{|x|^c}$

Réduction de l'erreur

La classe $\mathcal{BPP}$

Soit Σ un alphabet fini.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{BPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{1}{2} + \frac{1}{|x|^c}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] \geq \frac{1}{2} + \frac{1}{|x|^c}$

cf. le TD (*via* l'inégalité de Chernoff)

Classe de complexité $\mathcal{PP}$

La classe $\mathcal{PP}$

Soit Σ un alphabet fini. Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{PP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $O(|x|^k)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] > \frac{1}{2}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] > \frac{1}{2}$

$\mathcal{NP} \subseteq \mathcal{PP}$

Proposition

$$\mathcal{NP} \subseteq \mathcal{PP}$$

Démonstration. Il suffit de montrer que SAT $\in \mathcal{PP}$

- Soit Φ une formule booléenne en n variables $x_1, \ldots, x_n$
- Nous construisons une machine de Turing probabiliste de la façon suivante :
  - $(y_1, \ldots, y_n) \longleftarrow \{0,1\}^n$
  - Si $\Phi(y_1, \ldots, y_n) = 1$ alors retourner SATISFIABLE
  - Sinon $i \longleftarrow \{1, \ldots, 2^{n+1}\}$
    - si $i \leq 2^n - 1$ alors retourner SATISFIABLE
    - sinon retourner NON_SATISFIABLE

$\mathcal{NP} \subseteq \mathcal{PP}$

Démonstration (fin).

- Si Φ n'est pas satisfiable, l'algorithme retourne NON_SATISFIABLE avec probabilité

$$\frac{\#\{2^n, \ldots, 2^{n+1}\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{2^n + 1}{2^{n+1}} = \frac{1}{2} + \frac{1}{2^{n+1}} > \frac{1}{2}.$$

- Si Φ est satisfiable, il y a $t \geq 1$ assignation(s) $\vec{y}$ telles que $\Phi(\vec{y}) = 1$. L'algorithme retourne SATISFIABLE avec probabilité

$$\frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{\#\{1, \ldots, 2^n - 1\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{2^n - 1}{2^{n+1}}$$

$$= \frac{1}{2} - \frac{t}{2^{n+1}} - \frac{1}{2^{n+1}} + \frac{t}{2^n 2^{n+1}} + \frac{t}{2^n} > \frac{1}{2}$$

$\mathcal{NP} \subseteq \mathcal{PP}$

Démonstration (fin).

- Si Φ n'est pas satisfiable, l'algorithme retourne NON_SATISFIABLE avec probabilité

$$\frac{\#\{2^n, \ldots, 2^{n+1}\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{2^n + 1}{2^{n+1}} = \frac{1}{2} + \frac{1}{2^{n+1}} > \frac{1}{2}.$$

- Si Φ est satisfiable, il y a $t \geq 1$ assignation(s) $\vec{y}$ telles que $\Phi(\vec{y}) = 1$. L'algorithme retourne SATISFIABLE avec probabilité

$$\frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{\#\{1, \ldots, 2^n - 1\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{2^n - 1}{2^{n+1}}$$
$$= \frac{1}{2} - \frac{t}{2^{n+1}} - \frac{1}{2^{n+1}} + \frac{t}{2^n 2^{n+1}} + \frac{t}{2^n} > \frac{1}{2}$$

$\mathcal{NP} \subseteq \mathcal{PP}$

Démonstration (fin).

- Si Φ n'est pas satisfiable, l'algorithme retourne $\textsc{Non\_Satisfiable}$ avec probabilité

$$\frac{\#\{2^n, \ldots, 2^{n+1}\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{2^n + 1}{2^{n+1}} = \frac{1}{2} + \frac{1}{2^{n+1}} > \frac{1}{2}.$$

- Si Φ est satisfiable, il y a $t \geq 1$ assignation(s) $\vec{y}$ telles que $\Phi(\vec{y}) = 1$. L'algorithme retourne $\textsc{Satisfiable}$ avec probabilité

$$\begin{aligned}&\frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{\#\{1, \ldots, 2^n - 1\}}{\#\{1, \ldots, 2^{n+1}\}} = \frac{t}{2^n} + \frac{2^n - t}{2^n} \cdot \frac{2^n - 1}{2^{n+1}}\\ =&\frac{1}{2} - \frac{t}{2^{n+1}} - \frac{1}{2^{n+1}} + \frac{t}{2^n 2^{n+1}} + \frac{t}{2^n} > \frac{1}{2}\end{aligned}$$

$\square$

Table des matières

1. Définitions
   - Rappels
   - Machines de Turing probabilistes
2. Classe de complexité $\mathcal{BPP}$
   - Définition
   - Réduction de l'erreur et classe $\mathcal{PP}$
3. Classes de complexité $\mathcal{RP}$, $co-\mathcal{RP}$ et $\mathcal{ZPP}$
   - Classes de complexité $\mathcal{RP}$ et $co-\mathcal{RP}$
   - Classe de complexité $\mathcal{ZPP}$
4. Conclusion

La classe $\mathcal{RP}$

La classe $\mathcal{RTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \to \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{RTIME}(T)$ si et seulement si il existe une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $T(|x|)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] = 1$

La classe $\mathcal{RP}$

$$\mathcal{RP} = \bigcup_{k \in \mathbb{N}} \mathcal{RTIME}(n \longmapsto n^k)$$

La classe $\mathcal{RP}$

La classe $\mathcal{RTIME}$

Soit Σ un alphabet fini. Soit $T = \mathbb{N} \to \mathbb{N}$.
Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{RTIME}(T)$ si et seulement si il existe une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps au plus $T(|x|)$ (pour tous les tirages aléatoires)
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3}$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] = 1$

La classe $\mathcal{RP}$

$$\mathcal{RP} = \bigcup_{k \in \mathbb{N}} \mathcal{RTIME}(n \longmapsto n^k)$$

La classe $\mathcal{ZPP}$

La classe $\mathcal{ZPP}$

Soit Σ un alphabet fini. Un langage $\mathcal{L} \subseteq \Sigma^*$ appartient à la classe $\mathcal{ZPP}$ si et seulement si il existe un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps espéré $O(|x|^k)$
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] = 1$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] = 1$

Algorithme de type Las Vegas

La classe $\mathcal{ZPP}$

$$\mathcal{ZPP} = \mathcal{RP} \cap \text{co} - \mathcal{RP}$$

Démonstration.

- Montrons que $\mathcal{ZPP} \subseteq \mathcal{RP}$.

Soit $\mathcal{L}$ un langage de $\mathcal{ZPP}$. Par définition, il existe $T : \mathbb{N} \to \mathbb{N}$ avec $T(n) = O(n^k)$ pour un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

- $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps espéré $T(|x|)$
- Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] = 1$
- Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] = 1$

La classe $\mathcal{ZPP}$

$$\mathcal{ZPP} = \mathcal{RP} \cap \mathrm{co}-\mathcal{RP}$$

Démonstration.

- Montrons que $\mathcal{ZPP} \subseteq \mathcal{RP}$.

  Soit $\mathcal{L}$ un langage de $\mathcal{ZPP}$. Par définition, il existe $T : \mathbb{N} \to \mathbb{N}$ avec $T(n) = O(n^k)$ pour un entier k et une machine de Turing probabiliste $\mathcal{M}$ telle que

  - $\mathcal{M}$ termine toute exécution sur une entrée $x \in \Sigma^*$ en temps espéré $T(|x|)$
  - Pour tout $x \in \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 1] = 1$
  - Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons $\Pr[\mathcal{M}(x) = 0] = 1$

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construisons une machine $\mathcal{M}'$ qui sur l'entrée x
  - exécute $\mathcal{M}$ sur l'entrée x pour au plus $3 \cdot T(|x|)$ étapes
  - $\mathcal{M}$ s'arrête dans son état acceptant $\rightsquigarrow$ $\mathcal{M}'$ se place dans son état acceptant
  - $\mathcal{M}$ s'arrête dans un état non-acceptant $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant
  - $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant.
- $x \notin \mathcal{L} \rightsquigarrow \mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow \mathcal{M}'$ ne renvoie jamais 1 $\rightsquigarrow \Pr[\mathcal{M}'(x) = 0] = 1$
- $x \in \mathcal{L} \rightsquigarrow \mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.

Quelle probabilité ?

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construisons une machine $\mathcal{M}'$ qui sur l'entrée x
  - exécute $\mathcal{M}$ sur l'entrée x pour au plus $3 \cdot T(|x|)$ étapes
  - $\mathcal{M}$ s'arrête dans son état acceptant $\rightsquigarrow$ $\mathcal{M}'$ se place dans son état acceptant
  - $\mathcal{M}$ s'arrête dans un état non-acceptant $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant
  - $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant.

- $x \notin \mathcal{L}$ $\rightsquigarrow$ $\mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow$ $\mathcal{M}'$ ne renvoie jamais 1 $\rightsquigarrow$ $\Pr[\mathcal{M}'(x) = 0] = 1$

- $x \in \mathcal{L}$ $\rightsquigarrow$ $\mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.

  Quelle probabilité ?

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construisons une machine $\mathcal{M}'$ qui sur l'entrée x
  - exécute $\mathcal{M}$ sur l'entrée x pour au plus $3 \cdot T(|x|)$ étapes
  - $\mathcal{M}$ s'arrête dans son état acceptant $\rightsquigarrow$ $\mathcal{M}'$ se place dans son état acceptant
  - $\mathcal{M}$ s'arrête dans un état non-acceptant $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant
  - $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes $\rightsquigarrow$ $\mathcal{M}'$ s'arrête dans un état non-acceptant.

- $x \notin \mathcal{L} \rightsquigarrow \mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow \mathcal{M}'$ ne renvoie jamais 1 $\rightsquigarrow \Pr[\mathcal{M}'(x) = 0] = 1$

- $x \in \mathcal{L} \rightsquigarrow \mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.

  Quelle probabilité ?

Inégalité de Markov – Rappel

Inégalité de Markov

Soit Z une variable aléatoire réelle positive.

$$\forall a > 0, \qquad \Pr(Z \geqslant a) \leqslant \frac{\mathbb{E}(Z)}{a}.$$

Application.

- $Z =$ variable aléatoire du temps de $\mathcal{M}$ sur l'entrée x
- $\mathbb{E}(Z) \leq T(|x|) \rightsquigarrow \Pr\left(Z \geqslant 3T(|x|)\right) \leq \Pr\left(Z \geqslant 3 \cdot \mathbb{E}(Z)\right) \leqslant \frac{1}{3}$

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construis une machine $\mathcal{M}'$ avec dans le pire des cas $O(T(|x|)) = O(n^k)$ étapes ($+$ simulation du compteur)
- $x \notin \mathcal{L} \rightsquigarrow \mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow \mathcal{M}'$ ne renvoie jamais 1 $\rightsquigarrow \Pr[\mathcal{M}'(x) = 0] = 1$
- $x \in \mathcal{L} \rightsquigarrow \mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.
  $\rightsquigarrow \Pr[\mathcal{M}'(x) = 1] \geq 2/3$

Donc $\mathcal{L} \in \mathcal{RP}$ et $\mathcal{ZPP} \subset \mathcal{RP}$.

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construis une machine $\mathcal{M}'$ avec dans le pire des cas $O(T(|x|)) = O(n^k)$ étapes (+ simulation du compteur)
- $x \notin \mathcal{L} \rightsquigarrow \mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow \mathcal{M}'$ ne renvoie jamais 1
  $\rightsquigarrow \Pr[\mathcal{M}'(x) = 0] = 1$
- $x \in \mathcal{L} \rightsquigarrow \mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.
  $\rightsquigarrow \Pr[\mathcal{M}'(x) = 1] \geq 2/3$

Donc $\mathcal{L} \in \mathcal{RP}$ et $\mathcal{ZPP} \subset \mathcal{RP}$.

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Nous avons construis une machine $\mathcal{M}'$ avec dans le pire des cas $O(T(|x|)) = O(n^k)$ étapes ($+$ simulation du compteur)
- $x \notin \mathcal{L} \rightsquigarrow \mathcal{M}$ ne renvoie jamais 1 $\rightsquigarrow \mathcal{M}'$ ne renvoie jamais 1
$\rightsquigarrow \Pr[\mathcal{M}'(x) = 0] = 1$
- $x \in \mathcal{L} \rightsquigarrow \mathcal{M}'$ renvoie 0 seulement si $\mathcal{M}$ ne s'est pas arrêté au bout de $3 \cdot T(|x|)$ étapes.
$\rightsquigarrow \Pr[\mathcal{M}'(x) = 1] \geq 2/3$

Donc $\mathcal{L} \in \mathcal{RP}$ et $\mathcal{ZPP} \subset \mathcal{RP}$.

La classe $\mathcal{ZPP}$

Démonstration (suite).

- De même $\mathcal{ZPP} \subset \mathrm{co}-\mathcal{RP}$ et $\mathcal{ZPP} \subset \mathcal{RP} \cap \mathrm{co}-\mathcal{RP}$.

- Réciproquement, montrons que $\mathcal{RP} \cap \mathrm{co}-\mathcal{RP} \subset \mathcal{ZPP}$. Soit $\mathcal{L} \in \mathcal{RP} \cap \mathrm{co}-\mathcal{RP}$. Il existe deux machines de Turing probabilistes polynomiales $\mathcal{M}$ et $\mathcal{M}'$ telles que
  - Pour tout $x \in \mathcal{L}$, nous avons
  $$\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3} \text{ et } \Pr[\mathcal{M}'(x) = 1] = 1$$
  - Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons
  $$\Pr[\mathcal{M}(x) = 0] = 1 \text{ et } \Pr[\mathcal{M}'(x) = 0] \geq \frac{2}{3}$$

La classe $\mathcal{ZPP}$

Démonstration (suite).

- De même $\mathcal{ZPP} \subset \mathrm{co} - \mathcal{RP}$ et $\mathcal{ZPP} \subset \mathcal{RP} \cap \mathrm{co} - \mathcal{RP}$.
- Réciproquement, montrons que $\mathcal{RP} \cap \mathrm{co} - \mathcal{RP} \subset \mathcal{ZPP}$.
Soit $\mathcal{L} \in \mathcal{RP} \cap \mathrm{co} - \mathcal{RP}$. Il existe deux machines de Turing probabilistes polynomiales $\mathcal{M}$ et $\mathcal{M}'$ telles que
  - Pour tout $x \in \mathcal{L}$, nous avons
  $$\Pr[\mathcal{M}(x) = 1] \geq \frac{2}{3} \text{ et } \Pr[\mathcal{M}'(x) = 1] = 1$$
  - Pour tout $x \in \Sigma^* \setminus \mathcal{L}$, nous avons
  $$\Pr[\mathcal{M}(x) = 0] = 1 \text{ et } \Pr[\mathcal{M}'(x) = 0] \geq \frac{2}{3}$$

La classe $\mathcal{ZPP}$

Démonstration (suite).

- Si $\mathcal{M}(x) = 1$, alors $x \in \mathcal{L}$
- Si $\mathcal{M}'(x) = 0$, alors $x \notin \mathcal{L}$
- Si $\mathcal{M}(x) = 0$ et $\mathcal{M}'(x) = 1$, alors on ne peut rien déterminer de façon certaine mais ça n'arrive pas trop souvent ...

- On construit donc une machine $\mathcal{M}''$ qui
  - exécute $\mathcal{M}$ sur x, si $\mathcal{M}(x) = 1$ alors $\mathcal{M}''$ accepte
  - exécute $\mathcal{M}'$ sur x, si $\mathcal{M}'(x) = 0$ alors $\mathcal{M}''$ rejette
  - sinon $\mathcal{M}''$ recommence (avec probabilité $\leq 1/3$)

- $\mathcal{M}''$ ne se trompe jamais et s'exécute en temps espéré polynomial
  (la boucle est répétée en moyenne moins de $3/2$ fois)

Table des matières

1 Définitions
- Rappels
- Machines de Turing probabilistes

2 Classe de complexité $\mathcal{BPP}$
- Définition
- Réduction de l'erreur et classe $\mathcal{PP}$

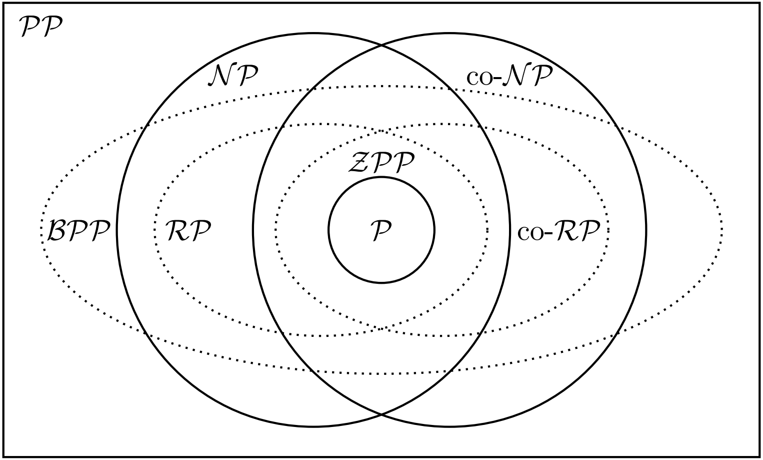
3 Classes de complexité $\mathcal{RP}$, $co-\mathcal{RP}$ et $\mathcal{ZPP}$
- Classes de complexité $\mathcal{RP}$ et $co-\mathcal{RP}$
- Classe de complexité $\mathcal{ZPP}$

4 Conclusion

Classes de complexité probabilistes

Classe	$\mathcal{M}(x) = 1 \| x \in \mathcal{L}$	$\mathcal{M}(x) = 0 \| x \notin \mathcal{L}$	temps
$\mathcal{P}$	1	1	pire cas
$\mathcal{ZPP}$	1	1	espéré
$\mathcal{RP}$	$\geq 2/3$	1	pire cas
co-$\mathcal{RP}$	1	$\geq 2/3$	pire cas
$\mathcal{BPP}$	$\geq 2/3$	$\geq 2/3$	pire cas
$\mathcal{NP}$	> 0	1	non dét.
co-$\mathcal{NP}$	1	> 0	non dét.
$\mathcal{PP}$	$> 1/2$	$> 1/2$	pire cas

Classes de complexité probabilistes



Classes de complexité probabilistes