Algos probabilistes

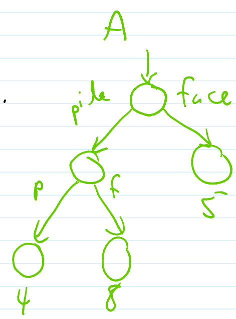
23 septembre 2021 10:07

Algo qui peut faire des choix selon une source de hasard.

- la sol. retournée dépend de ces choix, et donc APP est une var. aléatoire.
- on analyse APP en espérance. (expectation)

Soit A un algo probabiliste
Soit APP(A) la valeur de la sol. retournée par A. (var. aléatoire).
Pour une valeur numérique K, on dénote

$Pr[APP(A) = K]$ = proba. que A retourne une sol. de valeur K.



$Pr[APP(A) = 5] = \frac{1}{2}$
$Pr[APP(A) = 4] = 1/4$

L'espérance de A est la "moyenne" de K, pondérée par leur proba.

$$\mathbb{E}[APP(A)] = \sum_K K \cdot Pr[APP(A) = K]$$

- On dit que A est une c-approx. probabiliste si :
  - A s'exécute <u>toujours</u> en temps polynomial
  - si A résout un pb de minimisation
    $\forall$ instance, $\mathbb{E}[APP(A)] \leq c \cdot OPT$ $\quad c \geq 1$

    si A résout un pb de max.
    $\forall$ instance, $\mathbb{E}[APP(A)] \geq c \cdot OPT$ $\quad c \leq 1$

<u>Thm</u> : [linéarité de l'espérance]
soient $X_1, X_2, \ldots, X_n$ des vars. aléatoires,

$$\mathbb{E}\left[\sum_{i=1}^{n} X_i\right] = \sum_{i=1}^{n} \mathbb{E}[X_i]$$

$\frac{1}{2}$-approx proba. pour MAX-CUT

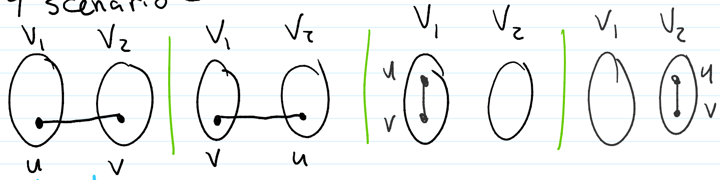
Idée : $\forall v \in V(G)$, mettre v dans V_1 avec prob $\frac{1}{2}$
V_2 avec prob $\frac{1}{2}$

Idée : $\forall v \in V(G)$, mettre v dans V_1 avec prob $\frac{1}{2}$
V_2 avec prob $\frac{1}{2}$

```
maxcutRando(G)
    V1 = ∅ , V2 = ∅
    pour v ∈ V(G)                          O(n)
        res = lancerPiece()   //pile = 1/2   face = 1/2
        si res = pile
        |   V1.add(v)
        sinon
        |   V2.add(v)
    return (V1, V2)
```

Intuition : $\forall uv \in E(G)$, uv a $\frac{1}{2}$ chance de traverser

4 scénarios



$\frac{1}{2} \cdot \frac{1}{2}$

$\underbrace{\frac{1}{4} \qquad \frac{1}{4}}_{\frac{1}{2} \text{ chance d'avoir } uv \text{ qui traverse}} \qquad \frac{1}{4} \qquad \frac{1}{4}$

Thm : maxcutRando est une $\frac{1}{2}$-approx proba.

Preuve : I_{uv} var. indicatrice indiquant si uv traverse

$\forall uv \in E(G)$, on définit

$$I_{uv} = \begin{cases} 1 & \text{si } u \in V_1, v \in V_2 \text{ ou } u \in V_2, v \in V_1 \\ 0 & \text{sinon} \end{cases}$$

Donc $APP = \sum_{uv \in E} I_{uv}$

$$\mathbb{E}[APP] = \mathbb{E}\left[\sum_{uv \in E} I_{uv}\right] = \sum_{uv \in E} \mathbb{E}[I_{uv}]$$

Que vaut $\mathbb{E}[I_{uv}]$?

Tel qu'argumenté plus haut, on a une chance sur deux que uv traverse (une chance une deux

Tel qu'argumenté plus haut, on a une chance sur deux que uv traverse (une chance une deux que uv ne traverse pas).

$$\Rightarrow \mathbb{E}[I_{uv}] = \frac{1}{2}\cdot 1 + \frac{1}{2}\cdot 0 = \frac{1}{2}$$

$$\Rightarrow \mathbb{E}[APP] = \sum_{uv\in E} \mathbb{E}[I_{uv}] = \sum_{uv\in E} \frac{1}{2} = \frac{1}{2}\cdot m \quad (m = |E(G)|)$$

Bien sûr, $OPT \leq m$

$$\Rightarrow \mathbb{E}[APP] = \frac{1}{2}\cdot m \geq \frac{1}{2}\cdot OPT. \quad \blacksquare$$

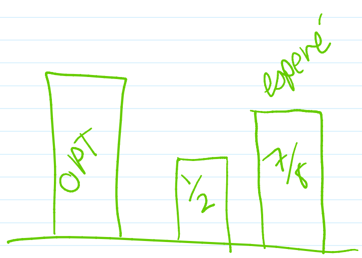
MAX-3-SAT

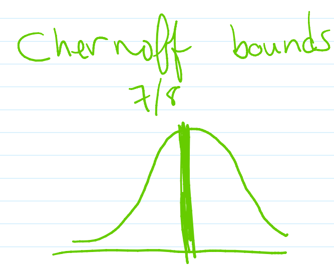
Entrée: clauses $C_1, \dots, C_m$ où chaque C_i a 3 variables

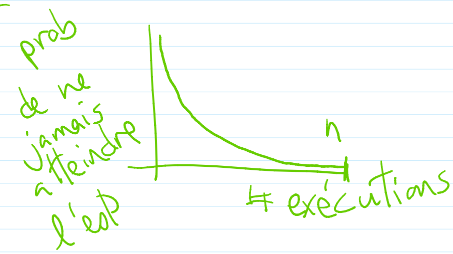
$C_i = x_1 \vee \overline{x}_2 \vee x_3$

Sortie: assignation qui sat. un # max de clauses

```
max3satProba( C = {C_1, ..., C_m}, sur variable x_1, ..., x_n)
    A = Ø
    pour i = 1..n
    |   res = lancerPiece()
    |   x_i = (res == pile)
    |   A += x_i
    x
    return A
```







Thm: max3satProba est une $7/8$-approx

Considérons une clause C_i avec variables x_1, x_2, x_3

ex: $C_i = x_1 \vee \overline{x}_2 \vee x_3$

Il y a une seule façon de ne pas satisfaire C_i

ex: $x_1 = F \quad x_2 = T \quad x_3 = F$

parmi $2^3 = 8$ assignations possibles de ses variables.

$\Rightarrow$ Il y a une prob. $\frac{1}{8}$ de ne pas satisfaire C_i

$\Rightarrow$ Prob $7/8$ de sat. C_i

$\Rightarrow$ Il y a une prob. $\frac{1}{8}$ de ne pas satisfaire C_i
$\Rightarrow$ Prob $7/8$ de sat. C_i

Soit $I_{C_i} = \begin{cases} 1 & C_i \text{ est satisf.} \\ 0 & \text{sinon} \end{cases}$

$$\mathbb{E}[APP] = \mathbb{E}\left[\sum_{i=1}^{m} I_{C_i}\right] = \sum_{i=1}^{m} \mathbb{E}[I_{C_i}] = \sum_{i=1}^{m} \frac{7}{8} = \frac{7m}{8}$$

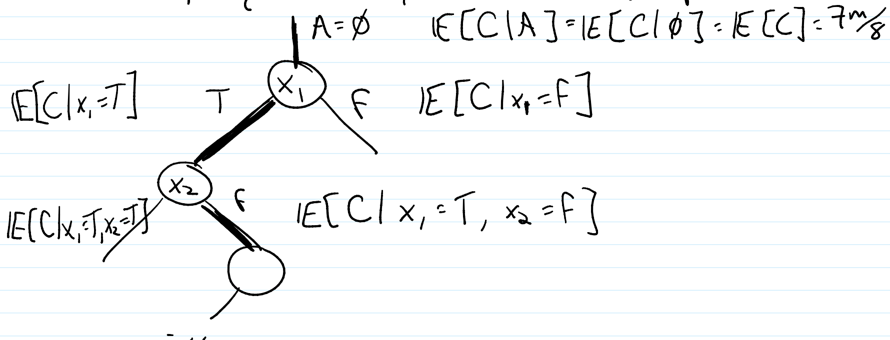
car $\mathbb{E}[I_{C_i}] = \frac{7}{8} \cdot 1 + \frac{1}{8} \cdot 0 = 7/8$

Puisque $OPT \leq m$, on a $APP \geq \frac{7}{8}\, OPT$ ■

Dérandomisation

(maxim.)
Proba $\rightarrow$ Déterministe t.q. $APP_{det} \geq \mathbb{E}[APP_{prob}]$

Idée : construire une solution un elt à la fois
à chaque étape, prendre le choix
qui garantit qu'on maintient l'espérance.



Soit A une assignation partielle.
Soit $C = \{C_1, \dots, C_m\}$
On dénote

$$\mathbb{E}[C \mid A] = \sum_{i=1}^{m} \Pr[C_i \text{ est satisfaite} \mid A]$$

par l'algo qui affecte les x_i restants

$$\Pr[C_i \text{ sat} \mid A] = \begin{cases} 0 & \text{si } A \text{ affecte déjà les vars de } C_i \text{ sans la satisfaire} \\ 1 & \text{si } A \text{ satisfait déjà } C_i \\ 1 - \frac{1}{2^k} & \text{où } k = \#\text{ de vars de } C_i \text{ non-assignées par } A \end{cases}$$

$C_i = x_1 \vee \overline{x_2} \vee x_3$

A: $\overline{x = F}$

$C_i = x_1 \vee x_2 \vee x_3$

$A: \overline{x}=f$

C_i non-assignées par A

$\Rightarrow$ On peut calculer $\mathbb{E}[C \mid A]$ en temps poly.

```
max3satDerando(C = {C_1, ..., C_m})
    A = ∅
    pour i = 1 .. n
        E_T = 0 , E_F = 0
        pour C_i ∈ C
            E_T += Pr[C_i | A + (x_i := T)]
            E_F += Pr[C_i | A + (x_i := F)]
        si E_T > E_F
            A = A + (x_i := T)
        sinon
            A = A + (x_i := F)
    return A
```

Ceci est une $\frac{7}{8}$-approx déterministe

À chaque étape, $\mathbb{E}[C \mid A] \geq \frac{7m}{8}$

Vrai quand $A = \emptyset$

Sinon, au début de l'iter i, $\mathbb{E}[C \mid A] \geq \frac{7m}{8}$

On a

$$\frac{7m}{8} \leq \mathbb{E}[C \mid A] = \frac{1}{2}\mathbb{E}[C \mid A + (x_i := T)] + \frac{1}{2}\mathbb{E}[C \mid A + (x_i := F)]$$

$\Rightarrow$ Un des deux choix garantit qu'on reste au-dessus de $\frac{7m}{8}$.