Schémas d'approximation et simulation.

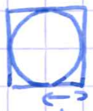
[Chap 10 du Mitzenmacher - Upfal Probability and computing]

a) (ε, δ) approximation.

P un problème qui a une solution de valeur V
A un algorithme qui renvoie X, une variable aléatoire.

$0 \leq \varepsilon, \delta \leq 1$ A est une (ε, δ) approximation pour la valeur V si
$$P(|X - V| \leq \varepsilon V) \geq 1 - \delta.$$

Exemple : approximation de π.



On place m points au hasard uniformément sur le carré, indépt.
$$Z_i = \begin{cases} 1 & \text{si le ième pt est dans le cercle} \\ 0 & \text{sinon.} \end{cases} \quad : P(Z_i = 1) = \frac{\pi}{4}.$$

$Z_1 \dots Z_m$ iid, $\sim \text{Ber}(\pi/4)$. Soit $Z = \sum_{i=1}^{n} Z_i$ $\quad E[Z] = \frac{m\pi}{4}$. $\quad W = \frac{4}{m} Z \; (E(W) = \pi)$

Borne de Chernoff $(\varepsilon < 1)$ $\quad P(|W - \pi| \geq \varepsilon\pi) = P(|Z - \frac{m\pi}{4}| \geq \varepsilon \frac{m\pi}{4}) \leq 2 e^{-\frac{m\pi\varepsilon^2}{12}}$

Pour avoir une (ε, δ) approx, il faut que $2e^{-m\pi\varepsilon^2/12} \leq \delta$ et donc $m \geq \frac{12 \ln(2/\delta)}{\pi\varepsilon^2}$

b) Schéma d'approximation polynomial

P un problème x entrée $\quad P(x)$ la solution pour cette entrée
A un algorithme x entrée $\quad A(x)$ la v.a. renvoyée par l'algo pour cette entrée.

FPRAS : fully polynomial randomized approximation scheme pour un pb P est un algo aléatoire A pour lequel $\forall \varepsilon, \delta$, $0 \leq \varepsilon, \delta \leq 1$, $\forall x$ entrée $A(x)$ est une (ε, δ)-approx de $V(x)$ et se calcule en un tps polynomial en $1/\varepsilon$, $\ln 1/\delta$ et la taille de x.

Lemme utile : Si $X_1 \dots X_m$ iid, $E(X_i) = \mu$. Si $m \geq \frac{3 \ln(2/\delta)}{\varepsilon^2 \mu}$ alors
$$P\left(\left|\frac{1}{m} \sum X_i - \mu\right| \geq \varepsilon\mu\right) \leq \delta.$$

Application au pb de comptage

Ex : formule en DNF (forme normale disjonctive) $\quad F = C_1 \vee C_2 \vee \dots \vee C_t \quad C_i = x_1 \wedge \bar{x}_2 \dots$

Problème : combien d'affectations des variables satisfont F ?

Le problème de décision associé est : existe-t-il une affectation des variables qui satisfait F ?
$\to$ problème facile : il suffit qu'une clause soit satisfiable et donc qu'une clause ne soit pas contradictoire.

Le problème difficile qui est associé est : CNF (forme normale conjonctive) qui est NP-complet $\to$ le problème de comptage associé à CNF est aussi difficile.

Mais alors le pb de comptage pour DNF est difficile :
CNF $\to$ DNF
$H \mapsto \bar{H}$ en utilisant les lois de DeMorgan

$$\underbrace{(\bar{x}_1 \wedge \bar{x}_2 \wedge \bar{x}_3) \vee (x_2 \wedge \bar{x}_4) \vee \dots}_{\bar{H}} \longleftarrow \underbrace{(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_2 \vee x_4)}_{H}$$

Classe de complexité #P : ∃ MT non det en tps polynomial tel que ∀ entrée x, le nbre de ch. acceptants est le nbre de solutions.

Plus pratiquement : il existe un algorithme (déterministe) qui vérifie qu'une instance du pb est solution en tps polynomial.
Si A est un pb NP alors #A est le pb de compter le nbre d'instances qui sont solutions de ce problème.

Autres exemples de pb de comptage difficiles
- nbre de couplages parfaits (alors que le pb de décision est facile)
- nbre de ch. Hamiltoniens

Algorithme naïf #DNF   entrée : Formule F

$X \leftarrow 0$
Pour k allant de 1 à m faire
. générer une affectation des variables de manière uniforme
. si cette affectation satisfait F alors $X \leftarrow X+1$
Renvoyer $Y \leftarrow \frac{X}{m} 2^n$.

On note $c(F)$ le nombre d'affectations qui satisfont F.

On a $E(Y) = \frac{2^n E(X)}{m} = c(F)$. $\left(E(X) = \frac{\overbrace{\#\text{clauses sat}}^{c(F)} \times m}{2^n}\right)$.

→ pour avoir une (ε, δ) approx, il faut $m \geqslant \frac{3.2^n}{\varepsilon^2 c(F)} \ln(2/\delta)$ → pas polynomial en la taille de l'entrée. exponentiel

Un FPRAS pour #DNF.

$F = C_1 \vee C_2 \vee \dots \vee C_t$. On suppose qu'aucune clause ne contient x_i et $\neg x_i$ (sinon on peut éliminer cette clause).

- Si C_i a l_i littéraux alors exactement 2^{n-l_i} affectations satisfont C_i
- On note $SC_i = \{x \mid F(x) = 1\}$ et $U = \{(i,x) \mid 1 \leq i \leq t, x \in SC_i\}$

$|SC_i| = 2^{n-l_i}$, $\sum_{i=1}^{t} |SC_i| = |U|$ et $c(F) = |\bigcup_{i=1}^{t} SC_i|$. $[c(F) \leq U]$.

Soit $S = \{(i,x) \mid 0 \leq i \leq t, x \in SC_i, x \notin SC_j \ \forall j < i\}$.
[pour chaque x tq $F(x) = 1$, on prend la 1ère clause qui est satisfaite uniqt.
Donc $|S| = c(F)$.

$|U|$ est connu, donc on peut estimer $c(F)$ en estimant $\frac{|S|}{|U|}$.

et comme $\frac{|S|}{|U|} \geqslant \frac{1}{t}$, S est a priori + dense dans U que l'ens. des solutions dans l'ensemble des affectations.

Algo : FPRAS pour #DNF

$X \leftarrow 0$
- Choisir i avec proba $\frac{|SC_i|}{\sum_{j=1}^{t} |SC_j|} = \frac{|SC_i|}{|U|}$
- Choisir x au hasard dans SC_i (l_i variables fixées, les autres au hasard)
- si $\forall j < i$ $x \notin SC_j$ alors $X \leftarrow X+1$
- Renvoyer $Y = \left(\frac{X}{m}\right) \sum_{i=1}^{t} |SC_i|$

On vérifie bien que $P((i,x) \text{ choisi}) = P(i \text{ choisi})\, P(x \text{ choisi} \mid i \text{ choisi})$

$$= \frac{|SC_i|}{|U|} \times \frac{1}{|SC_i|} = \frac{1}{|U|}$$

donc $E(Y) = \frac{E(X)}{m} \sum |SC_i| = \frac{m\, c(F)}{\sum |SC_i|} \times \frac{\sum SC_i}{m} = c(F)$.

De plus, on peut choisir $m = \lceil 3t/\varepsilon^2 \ln\left(\frac{2}{\delta}\right) \rceil$ car $E(X) \geq \frac{m}{t}$.

C) Lien entre échantillonnage et approximation du comptage.

Déf: Un échantillonneur ε-uniforme est un algorithme qui génère $\omega \in \Omega$ tel que $\forall S \subseteq \Omega \quad \left| P(\omega \in S) - \frac{|S|}{|\Omega|} \right| \leq \varepsilon$

(Lien direct avec la distance de variation).

FPAUS : fully polynomial almost uniform sampler pour Ω est un algo qui pour l'entrée x $\forall \varepsilon > 0$, il génère un échantillon ε-uniforme de $\Omega(x)$ et s'exécute en tps polynomial en $\ln(1/\varepsilon)$ et en la taille de x.

But : transformer un FPAUS en un FPRAS.

Exemple : dénombrer les indépendants d'un graphe. [comme dans l'exemple préc, on ne peut pas utiliser une méthode naïve : 2^n ss-ens de sommets].

$G = (V, E)$ graphe non-orienté

$E = \{e_1, \dots e_m\}$ $\quad G_m = G, \quad G_{i-1} = G_i - \{e_i\}$.

$\Omega(G_i) \triangleq \{\text{indépendants de } G_i\}$

On a : $|\Omega(G)| = \frac{|\Omega(G_m)|}{|\Omega(G_{m-1})|} \times \frac{|\Omega(G_{m-1})|}{|\Omega(G_{m-2})|} \times \dots \times \frac{|\Omega(G_1)|}{|\Omega(G_0)|} \times \underbrace{|\Omega(G_0)|}_{=2^n}$.

On cherche à estimer $r_i = \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|}$. Si $\tilde{r}_i$ est un estimateur de r_i, alors $|\Omega(G)|$ est estimé par $2^n \prod \tilde{r}_i = \tilde{\Omega}$

L'erreur est de $R = \prod_{i=1}^{m} \frac{\tilde{r}_i}{r_i}$.

On veut obtenir $P(|\tilde{\Omega} - |\Omega(G)|| \leq \varepsilon |\Omega(G)|) = P\left(\left|\frac{\tilde{\Omega}}{|\Omega(G)|} - 1\right| \leq \varepsilon\right) = P(|R-1| \leq \varepsilon) \geq 1-\delta$.

Lemme: si $\tilde{r}_i$ est une $(\varepsilon/2m, \delta/m)$-approximation de r_i, alors $P(|R-1| \leq \varepsilon) \geq 1-\delta$.

Dém: On suppose que $\forall i \; P(|\tilde{r}_i - r_i| \leq \frac{\varepsilon}{2m} r_i) \geq 1 - \delta/m$. Donc $P(|\tilde{r}_i - r_i| > \frac{\varepsilon}{2m} r_i) < \delta/m$

Union bound: $P(\exists i \, |\tilde{r}_i - r_i| > \frac{\varepsilon}{2m} r_i) < \delta$ et donc $P(\forall i \; |\tilde{r}_i - r_i| \leq \frac{\varepsilon}{2m} r_i) \geq 1-\delta$.

Or $P(\forall i \; |\tilde{r}_i - r_i| \leq \frac{\varepsilon}{2m} r_i) \leq P\left(\left(1 - \frac{\varepsilon}{2m}\right)^m \leq \prod_i \frac{\tilde{r}_i}{r_i} \leq \left(1 + \frac{\varepsilon}{2m}\right)^m\right)$

$$\leq P\Big(1 - \varepsilon \leq \underbrace{\prod \frac{\tilde{r}_i}{r_i}}_{R} \leq 1+\varepsilon\Big).$$

Donc $P(|R-1| \leq \varepsilon) \geq 1-\delta$.

Algorithme: estimer r_i (on suppose que l'on dispose d'un FPAUS pour les indépendants).

Entrée: $G_{i-1} = (V, E_{i-1})$, $G_i = (V, E_i)$

Sortie : $\tilde{r}_i$

$X \leftarrow 0$

Répéter $n = \lceil \frac{1296\, m^2}{\varepsilon^2} \ln(2m/\delta) \rceil$ fois

a) générer un $\varepsilon/6m$-échantillon de $\Omega(G_{i-1})$

b) si c'est un indépendant de G_i alors $X \leftarrow X+1$

Renvoyer $\tilde{r}_i := X/n$.

Lemme : cet algorithme renvoie une $(\varepsilon/2m, \delta/m)$ approximation de r_i.

Dém : • $1/2 \leq r_i \leq 1$: $\quad f : \Omega(G_{i-1}) \longrightarrow \Omega(G_i) \quad (e_i = \{u,v\})$

$I \longmapsto I$ si $(u,v) \notin I^2$

$I \longmapsto I \setminus \{u,v\}$ si $(u,v) \in I^2$

$|f^{-1}(I)| \leq 2 \quad (f^{-1}(I) = \{I, I \cup \{u,v\}\})$

Donc $|\Omega(G_{i-1})| \leq 2 |\Omega(G_i)|$

• Soit $X_k = 1$ si le kè échantillon est un indép de G_i, et 0 sinon.

$\left| P(X_k = 1) - \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|} \right| \leq \frac{\varepsilon}{6m}$ Donc $\left| E(X_k) - \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|} \right| \leq \frac{\varepsilon}{6m}$.

Mais alors $|E[\tilde{r}_i] - r_i| = \left| E\left(\frac{\sum X_k}{n}\right) - \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|} \right| = \left| E[X_k] - \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|} \right| \leq \varepsilon/6m$.

• $E(\tilde{r}_i) \geq r_i - \varepsilon/6m \geq \frac{1}{2} - \frac{\varepsilon}{6m} \geq \frac{1}{2} - \frac{1}{6} = \frac{1}{3}$.

Donc on peut choisir $n \geq \frac{3 \ln(2m/\delta)}{1/3 \, (\varepsilon/12m)^2} = 1296\, m^2 \varepsilon^{-2} \ln\left(\frac{2m}{\delta}\right)$ alors

$P\left(\left|\frac{\tilde{r}}{E(\tilde{r}_i)} - 1\right| \geq \frac{\varepsilon}{12m}\right) = P\left(|\tilde{r}_i - E(\tilde{r}_i)| \geq \frac{\varepsilon}{12m} E(\tilde{r}_i)\right) \leq \frac{\delta}{m}$.

Donc $P\left(1 - \frac{\varepsilon}{12m} \leq \frac{\tilde{r}_i}{E(\tilde{r}_i)} \leq 1 + \frac{\varepsilon}{12m}\right) \geq 1 - \delta/m$

$1 - \frac{\varepsilon}{3m} \leq 1 - \frac{\varepsilon}{6mr_i} \leq \frac{E(\tilde{r}_i)}{r_i} \leq 1 + \frac{\varepsilon}{6mr_i} \leq 1 + \frac{\varepsilon}{3m} \quad \left(r_i \geq \frac{1}{2}\right)$

Donc $P\left(\left(1 - \frac{\varepsilon}{12m}\right)\left(1 - \frac{\varepsilon}{3m}\right) \leq \frac{\tilde{r}_i}{r_i} \leq \left(1 + \frac{\varepsilon}{12m}\right)\left(1 + \frac{\varepsilon}{3m}\right)\right) \geq 1 - \frac{\delta}{m}$

$\wedge|$

$P\left(\left(1 - \frac{\varepsilon}{2m}\right) \leq \frac{\tilde{r}_i}{r_i} \leq 1 + \frac{\varepsilon}{2m}\right) \geq 1 - \delta/m$.

Th : Si on dispose d'un FPAUS pour les ens. indépendants d'un graphe, alors on peut construire un FPRAS pour les ens indépendants.

Th (Jerrum, Valiant, Vazirani) : Tout pb #P-complet a soit un FPRAS soit est impossible à approximer.

[Pb #P complet si : • Pb $\in$ #P
• $\forall$ Pb' $\in$ #P, il existe un algo polynomial (déterministe) tq Pb' se réduit à Pb.