⚠ À chercher : 1/ → 3/2 dual approx pour Bin packing
2/ → Rq no FPTAS pour MKP SubsetSum avec gap red.

MKP :
C sac à dos de taille $a_1, \dots, a_j$
n objets { s_j size, p_j : profit

& part

placer les objets dans les C sacs en maximisant la $\sum_{j \text{ placés}} p_j$.

Résultats Négatifs

Utiliser la NP-hardness (⇒ pour "no FPTAS").

Lemme 1 : Si problème à valeurs entières, FPTAS ⇒ résolution (exacte) en poly (OPT(I)).

PROOF : Avec FPTAS, $\forall \varepsilon$, je peux avoir $A \leq (1+\varepsilon)\,OPT(I)$ en $poly\left(\frac{1}{\varepsilon}, n\right)$

⇒ pour avoir une solution exacte, il faut que $A \leq OPT(I) + \underbrace{\varepsilon\, OPT(I)}_{<1}$

$\varepsilon < \frac{1}{OPT(I)}$ donc lancer le FPTAS avec un tel ε formant une solution exacte en poly (OPT(I)).

Corollaire : Soit π pb de minimisation.

a) Si π est NP hard et OPT(I) poly borné ⇒ NO FPTAS (unless P=NP)
(par ex : $\max p_j \leq poly(n)$).

Si π est NP-hard sens fort et OPT(I) est pseudo poly borné ⇒ no FPTAS (unless P=NP).

b) Si on avait FPTAS, on aurait résolution exacte en poly (OPT(I)) de en poly(poly(max valeurs entrées), n)
donc appliqué à $\tilde{\pi} = \pi$ avec max valeurs $\leq poly(n)$, on aurait algo poly pour $\tilde{\pi}$, qui est NP-hard □.

Utiliser gap réductions.

Def (gap$_r$) : Soit π problème (minimisation). Soit $r : \mathcal{I} \to \mathbb{R}$, $I \mapsto r(I)$.

gap$_r$ est le problème suivant :
- input : I, k
- output : yes si $OPT(I) \geq k$, no si $OPT(I) \leq \frac{k}{r(I)}$

ce qu'on aurait sinon : oui
k
I →
ce qu'on veut
$k/r(I)$
non