Knapsack:

Input: - Items $[n]$

- profits $p : [n] \to \mathbb{N}$
- sizes $s : [n] \to \mathbb{N}$
- integer $k \in \mathbb{N}$

Feasible solution: $I \subseteq [n]$ s.t. $s(I) = \sum_{i \in I} s(i) \leq k$

Objective: $\max p(I) = \sum_{i \in I} p(i)$

WLOG: $s(i) \leq k \quad \forall i$

Sort: WLOG, assume $\frac{p(i)}{s(i)} \geq \frac{p(i+1)}{s(i+1)} \quad \forall i \in [n-1]$

Greedy:

- Add items until can't anymore

Bad!

- $s(1) = 1 \qquad p(1) = 2$
- $s(2) = k \qquad p(2) = k$

$\Rightarrow \Omega(k)$-approx

Improved greedy:

- Let i be first element we can't add:

$$\sum_{j=1}^{i} s(j) > k, \text{ but } \sum_{j=1}^{i-1} s(j) \le k$$

- Return better of $[i-1]$, $\{i\}$

<u>Thm</u>: 2-approximation

<u>Pf</u>:

Since greedy, know $\dfrac{\sum_{j=1}^{i} p(j)}{\sum_{j=1}^{i} s(j)} \ge \dfrac{OPT}{k}$ (greedy has better "bang for the buck")

$$\Rightarrow OPT \le \frac{k}{\sum_{j=1}^{i} s(j)} \cdot \sum_{j=1}^{i} p(j) \le \sum_{j=1}^{i} p(j)$$

$$\le \sum_{j=1}^{i-1} p(j) + p(i)$$

$$\le 2 \max\left(\sum_{j=1}^{i-1} p(j), p(i)\right)$$

$$= 2 \cdot ALG$$

$$\Rightarrow ALG \ge \frac{1}{2} \cdot OPT$$

Dynamic Programming:

$M = \max_{i \in [n]} p(i) \Rightarrow OPT \le nM$

Subproblems:

- What size is necessary to achieve some profit?

$\forall i \in [n]$, $0 \le v \le nM$, let $f(i,v)$ be min size necessary to achieve profit <u>exactly</u> v using elements in $[i]$

If could compute this, could solve knapsack!

$OPT = \max v$ s.t. $f(n,v) \le K$

Recurrence for $f(i,v)$:

$$f(i,v) = \begin{cases} 0 & i=0,\ v=0 \\ \infty & i=0,\ v>0 \\ f(i-1,v) & i>0,\ p(i)>v \\ \min(f(i-1,v),\ s(i)+f(i-1,v-p(i))) & i>0,\ p(i)\le v \end{cases}$$

Running time:

time / table entry: $O(1)$

table entries: $n \cdot nM = n^2 M$

Pseudopolynomial!

Speed this up: "round" the instance to make M small

Let $\varepsilon > 0$. We'll try for $(1-\varepsilon)$-approximation

Let $\delta = \frac{\varepsilon M}{n}$

Let $p'(i) = \left\lfloor \frac{p(i)}{\delta} \right\rfloor = \left\lfloor \frac{n}{\varepsilon} \cdot \frac{p(i)}{M} \right\rfloor$

"scales" so max profit $\frac{n}{\varepsilon}$, "rounds" down to integer

Alg:

- Run old alg on p' profits!

Running time:

$$O(n^2 \tfrac{n}{\varepsilon}) = O(\tfrac{n^3}{\varepsilon})$$

Thm: $(1-\varepsilon)$-approximation

Pf: Let $I \subseteq [n]$ returned by algorithm,

$I^* \subseteq [n]$ optimal solution

Since did not round sizes, I feasible

$p'(i) = \left\lfloor \frac{p(i)}{\delta} \right\rfloor$

Def of $p'(i)$: - $p(i) \geq \delta \cdot p'(i)$

- $p'(i) \geq \frac{p(i)}{\delta} - 1$

$$\begin{aligned}
ALG &= \sum_{i \in I} p(i) \\
&\geq \delta \sum_{i \in I} p'(i) \\
&\geq \delta \sum_{i \in I^*} p'(i) && (I \text{ is optimal for } p') \\
&\geq \delta \sum_{i \in I^*} \left(\frac{p(i)}{\delta} - 1 \right) \\
&= \sum_{i \in I^*} p(i) - \delta |I^*| \\
&\geq OPT - \delta n && (|I^*| \leq n) \\
&= OPT - \varepsilon M && \left(\delta = \frac{\varepsilon M}{n}\right) \\
&\geq OPT - \varepsilon \cdot OPT && (M \leq OPT) \\
&= (1 - \varepsilon) \cdot OPT
\end{aligned}$$

~~Fully~~ Polynomial Time Approximation Scheme (FPTAS):

$(1-\varepsilon)$-approximation, running time poly in $n, \frac{1}{\varepsilon}$

poly(n) for constant ε

$n^{1/\varepsilon}$ $2^{1/\varepsilon}$

Min-Makespan on Identical Parallel Machines

Input: - Jobs J $(|J| = n)$

- Machines M $(|M| = k)$
- Processing times $p : J \to \mathbb{R}$

Feasible Solution: $\Phi : J \to M$

Objective: min makespan $= \min \max_{m \in M} \sum_{j \in J : \Phi(j) = m} p(j)$