R07922118 資工碩一 翁子騰

Problem 4

(a)

先依照$\frac{v}{w}$大小排序，由大排到小(最大的還是編號設為 1，但是和題目的編號不同)。假設第 j 個是此 algorithm 拒絕的 item。

假設 0/1 knapsack 的 optimal solution 為 OPT, 用此 algorithm 的解為 A，

fractional knapsack 的 optimal solution 為$OPT_{fraction}$

要證明: $2A \geq OPT$

$A = \max\{\sum_{i=1}^{j-1} v_i, v_{max}\}$, 其中$v_{max} \geq v_j$

由於 fractional knapsack 不可能比 0/1 knapsack 的 optiaml solution 還差，可知

$$OPT_{fraction} \geq OPT \ldots\ldots (1)$$

$$且 OPT_{fraction} = \sum_{i=1}^{j-1} v_i + \alpha * v_j, where\ \alpha = \frac{W - \sum_{i=1}^{j-1} w_i}{w_j}$$

$OPT_{fraction} \leq \sum_{i=1}^{j-1} v_i + v_j \leq 2 * A$......(2)

Why?

如果 $\sum_{i=1}^{j-1} v_i > v_j$且$\sum_{i=1}^{j-1} v_i > v_{max}$ ，則$A = \sum_{i=1}^{j-1} v_i > v_j$ →

$$\sum_{i=1}^{j-1} v_i + v_j < \sum_{i=1}^{j-1} v_i + \sum_{i=1}^{j-1} v_i = 2 * A$$

如果 $\sum_{i=1}^{j-1} v_i > v_j$且$\sum_{i=1}^{j-1} v_i \leq v_{max}$ ，則$A = v_{max}$ →

$$\sum_{i=1}^{j-1} v_i + v_j < \sum_{i=1}^{j-1} v_i + \sum_{i=1}^{j-1} v_i \leq v_{max} + v_{max} = 2 * A$$

如果$v_{max} \geq v_j \geq \sum_{i=1}^{j-1} v_i$ ，$A = v_{max}$

$$\sum_{i=1}^{j-1} v_i + v_j \leq v_j + v_j = 2v_j \leq 2v_{max} = 2 * A$$

由(1), (2)可知$2A \geq OPT$，所以此$algorithm$為 $2 - approximation\ algorithm$。

(b)

DP (i, V) is the minimum weight with total value at least V within the first i items.

Recursive formula: →in small integer value this will give an exact solution!

$DP(i, V) = \min\big(DP(i-1, V),\ w_i + DP(i-1, V - v_i)\big)$沒選第 i 個 item 和有選第 i 個 item

$$DP(0, v) = \infty \; where \; 1 \le v \le V^*$$
$$DP(0,0) = 0$$

$DP(0,0) = 0$

$for \; v = 1 \; to \; V^*$: $DP(0, v) = \infty$

for i from 1 to n //total n items, O(n)

    for each possible v from 1 to V^* $//O(V^*)$

        $DP(i, V) = \min\big(DP(i-1, V),\; w_i + DP(i-1, V - v_i)\big)$

for k from $V^* \; to \; 1$ $// \; O(V^*)$

    if DP(n, k)<=W return k

填表格有兩個 for loop 花 O(n)∗ $O(V^*)$，最後找答案一個 for loop 花$O(V^*)$，一共花 O($n * V^*$)

(c)

1 找一個 K，並且把所有的 value 都 reduce 成 floor(v/K)，形成 v’ array //O(n)

2 利用新的 v’，代入(b)的 dynammic programming algorithm 算出 ans //

3 return ans*K

因為所有的 value 都被 reduced 過，所以算出來的 optimal value 也會接近$\frac{V^*}{K}$，所以第 2 步算 DP table 的時間複雜度會變為 O($n * (\frac{V^*}{K})$)。第 1 步算 reduced value 花 O(n)，所以一共花 O(n)+O($n * (\frac{V^*}{K})$)= O($n * (\frac{V^*}{K})$)。

(d)

假設 S*為原本 0/1 knapsack problem 的最佳解集合，S’*為 0/1 knapsack problem with reduced values 的最佳解集合，c(S*)為 S*用原 value 求出的總值，為原本問題中的 optimal value，c’(S*)為 S*中用 reduced values 求出的總值。

如此 c(S*)=V*, c’(S’*)= $\hat{V}$

由(c)所傳回的解為 K* c’(S’*)

$$c'(S'^*) \ge c'(S^*) \dots \dots (1)$$

因為我們假設$c'(S'^*)$為0/1 knapsack problem with reduced values 的 optimal value，至少會比其他解的 value 大，包含S^*。

由(1)，可推得

$$-K * c'(S'^*) \le -K * c'(S^*)$$
$$c(S^*) - K * c'(S'^*) \le c(S^*) - K * c'(S^*) \dots \dots (2)$$

其中$c(S^*) - K * c'(S^*) = \sum_{i\epsilon S^*} v_i - K * \sum_{i\epsilon S^*} \left\lfloor \frac{v_i}{K} \right\rfloor = \sum_{i\epsilon S^*}(v_i - K * \left\lfloor \frac{v_i}{K} \right\rfloor) <$

$\sum_{i\in S^*} K \leq n * K$

Why?

$$\frac{v_i}{K} - 1 < \left\lfloor \frac{v_i}{K} \right\rfloor \leq \frac{v_i}{K}$$

$$\rightarrow -K\left(\frac{v_i}{K} - 1\right) > -K * \left\lfloor \frac{v_i}{K} \right\rfloor \geq -K * \frac{v_i}{K}$$

$$\rightarrow v_i - K\left(\frac{v_i}{K} - 1\right) > v_i - K * \left\lfloor \frac{v_i}{K} \right\rfloor \geq v_i - K * \frac{v_i}{K}$$

$$\rightarrow K > v_i - K * \left\lfloor \frac{v_i}{K} \right\rfloor \geq 0$$

$$\therefore\ c(S^*) - K * c'(S'^*) \leq c(S^*) - K * c'(S^*) \leq n * K \ldots\ldots (3)$$

又已知 c(S*)=V*, c'(S'*)= $\hat{V}$，代入(3)，得

$$c(S^*) - K * c'(S'^*) = V^* - K * \hat{V} \leq n * K$$

$$V^* - n * K \leq K * \hat{V} \ldots\ldots (4)$$

又V^*是原問題最佳解的總值，$K * \hat{V}$

為我們利用$reduced\ value$重新估算的總值。可得$V^* \geq\ K * \hat{V} \ldots\ldots (5)$

由(4), (5)可得

$$V^* - n * K \leq K * \hat{V} \leq V^*$$

證明完畢。