Algorithms Design and Analysis [ETCS-301]

Dr. A K Yadav
Amity School of Engineering and Technology
(affiliated to GGSIPU, Delhi)
akyadav1@amity.edu
akyadav@akyadav.in
www.akyadav.in
+91 9911375598

October 16, 2019

Fractional Knapsack problem

- There are n item.
- Value of i^{th} item is v_i and weight is w_i.
- Knapsack capacity of the bag is W.
- We wish to maximize the total bag value subject to the constraint that total weight is less than or equal to W that is:

$$\text{Maximize } \sum_{i=1}^{n} v_i x_i$$

$$\text{under the constraints } \sum_{i=1}^{n} x_i w_i \leq W, \text{ and } x_i = [0, 1]$$

Optimal-substructure property

- Consider that W is the optimal load of the bag.
- If we remove a weight w of one item j from the optimal load, the remaining load must be the most valuable load weighing at most $W - w$ from the $n - 1$.
- If this $W^1 = W - w$ is not optimal but W^2 is optimal then $W^2 + w$ will be optimal not W
- But W is optimal under assumption so W^1 will be optimal.
- So Fractional Knapsack problem satisfied optimal-substructure property

Greedy property

- Suppose items are arranged in descending order of $\frac{v_i}{w_i}$
- Take W is the sum of weights from first i weights.
- If we remove a weight w of one item i from the optimal load, the remaining load must be the most valuable load weighing at most $W - w$ from the $i - 1$.
- Because weights are arranged in descending order of value per unit weight
- So if we remove one item then remaining items also more valuable then the rest.
- Hence greedy-choice property is applicable.

Correctness of the algorithm

- Suppose items are arranged in descending order of $\frac{v_i}{w_i}$
- Take W is the sum of weights from first i weights with value V.
- If we remove a weight w of one item $j \leq i$ from the optimal load and replace $w^{'}$ from the rest of item $k > i$
- The value of resultant load $V^{'} = V - v_j + v_k$
- But $V^{'} \leq V$ as $v_j \geq v_k$.
- So the the selection of the most valuable item is correct.

An iterative greedy algorithm

FKP(v, w, n, W)

1. Let items are arranged in descending order of $\frac{v_i}{w_i}$
2. J=0,V=0,i=1
3. while($i \leq n$ and $J < W$)
4. if $w_i \leq W - J$
5. $V = V + v_i$
6. $J = J + w_i$
7. else
8. $V = V + \frac{W-J}{w_i} \times v_i$
9. $J = W$
10. $i = i + 1$
11. return V

Complexity of the algorithm is $O(n \lg n)$ for sorting and $O(n)$ for selecting the weights. So it will be $O(n \lg n)$

Thank you

Please send your feedback or any queries to akyadav1@amity.edu, akyadav@akyadav.in or contact me on +91 9911375598