Approximation Algorithm for the Knapsack Problem

Hanan Ayad

Enhanced Greedy Algorithm

The algorithm can be described by the following steps.

1. Sort the n items such that $v_1/w_1 \geq v_2/w_2 \ldots \geq v_n/w_n$
2. Fill knapsack sequentially with items in above order until there are no more items left or an item k cannot fit, which means that the following condition holds,
$$w_k > W - \sum_{i=1}^{k-1} w_i \tag{1}$$
3. At this point, choose the best of the following two possibilities

   If $v_k < \sum_{i=1}^{k-1} v_i$, keep items 1 to $k-1$ in knapsack,

   else, replace all items 1 to $k-1$ with item k (ties can be broken arbitrarily).

This algorithm does not seem particularly clever, but it represents an approximation algorithm with guarranteed accuracy ratio ≤ 2. That is, $f(s^*)/f(s_a) \leq 2$ (as defined for maximization problems).

Proof

To show this, we consider the continous (fractional) knapsack problem, which unlike the discrete knapsack, can be solved optimally in polynomial-time by a simple greedy algorithm. In particular, the optimal solution to an instance of the knapsack problem can serve as an upper bound on the optimal value of the discrete version for the same instance.

The optimal solution computed by a simple greedy algorithm for the fractional knapsack (see algorithm description in textbook) can be written as follows.

$$\sum_{i=1}^{k-1} v_i + \frac{W - \sum_{i=1}^{k-1} w_i}{w_k} v_k \tag{2}$$

The optimal solution $f(s^*)$ of the discrete version for the same instance must be bounded from above by the expression in (2). So, from (1) and (2), we have,

$$f(s^*) \leq \sum_{i=1}^{k-1} v_i + v_k \tag{3}$$

The enhanced greedy algorithm returns the approximate solution $f(s_a)$ as $\max\{\sum_{i=1}^{k-1} v_i, v_k\}$. Substituting in (3), we get,

$$f(s^*) \leq 2f(s_a)$$