Discussion 9

Out: April 21, 2022 Discussed: April 22, 2022

1 Subset-Sum: DP approach

Consider the following **Subset-Sum** optimization problem:

- Input: A set of n non-negative integers $\{w_1, w_2, \ldots, w_n\}$, and a non-negative integer W.
- Output: the largest value $\sum_{w_i \in S} w_i$ among all subsets $S \subseteq \{w_1, \ldots, w_n\}$ subject to the constraint $\sum_{w_i \in S} w_i \leq W$.

Design a dynamic programming algorithm to solve this problem:

(i) Define a dynamic programming table T, and describe what each entry of the table corresponds to. What is the final answer of the algorithm in terms of the table entries?

(ii) Write a recurrence relation for entries of T, including base cases.

(iii) Provide pseudocode showing how to compute entries of T.

(iv) Analyze the running time of the algorithm? Is this polynomial time?

2 Subset-Sum: Hardness

Now consider the following **decision version** of the **Subset-Sum** problem:

- Input: A set of n non-negative integers $\{w_1, w_2, \ldots, w_n\}$, and a non-negative integer W.
- Output: True if there exists a subset $S \subseteq \{w_1, \ldots, w_n\}$ such that $\sum_{w_i \in S} w_i = W$, and False otherwise.

Consider also the following generalization of the Bipartite Matching problem, called **3D-Matching**:

- Input: three disjoint sets X, Y, Z each of size n, and a set $T \subseteq X \times Y \times Z$ of m ordered triples.
- Output: True if there exists a set of n triples in T such that each element of $X \cup Y \cup Z$ is contained in exactly one of these triples (i.e., a perfect three-dimensional matching) and False otherwise.

Assume that 3D-Matching is NP-Complete. Prove that Subset-Sum is NP-Hard by showing

$$\text{3D-Matching} \leq_p \text{Subset-Sum}.$$