F번: 총통의 축지법

문제를 잘 해석하면 임의의 $p_1 \in \{1, ..., N\}^M$ 에 대하여 다음 조건을 만족하는 집합 S를 찾으라는 문제이다.

여기에서 임의의 $P, Q \in \{1, ..., N\}^M$ 에 대하여 $P \oplus Q = (P_1 \oplus Q_1, ..., P_M \oplus Q_M)$ (단, 임의의 $a, b \in \mathbb{N}$에 관하여 $a \oplus b$는 a, b에 관한 XOR 연산이고, P_i, Q_i는 각각 $P, Q \in \{1, ..., N\}^M$ 에 대한 i번째 성분을 의미)로 정의한다.

- S는 총통이 축지법을 쓰면서 방문한 모든 정점들의 집합이다.
- 모든 $T \subset S$에 관하여 $\bigoplus_{p_i \in T} p_i$의 각 성분이 하나라도 0이 되지 않는다. (단, $T \neq \emptyset$)
- $\sum_{p_i \in S} d(p_1, p_i)$가 최대가 되는 집합이다.

이걸 대체 어떻게 풀 것인지 한참 고민하고 있던 당신에게 Matroid라는 것이 머리에 떠올랐다면 이 문제를 이미 풀었을 것이다. 각 행을 독립적으로 생각해보자. 그러면 부분 문제로 Vector matroid의 Maximum weight independent set의 크기를 구하는 것으로 볼 수 있다.

따라서 각 성분별로 독립적으로 부분문제를 풀어서 각 성분 중에서 independent set의 크기를 출력하면 이 문제가 풀릴 것이다. (Matroid의 성질에 의해서 임의의 independent set의 부분집합 또한 independent set 이다) 그런데, 어떻게 임의의 Matroid에 관하여 Maximum weight independent set을 구할까? 이 문제는 놀랍게도 Greedy 알고리즘으로 해결할 수 있다.

임의의 Matroid $\mathcal{M} = (E, \mathcal{I})$와 weight function $w : E \to \mathbb{R}_+$를 정의하자. 그리고 임의의 $T \subset E$가 independent set인지 판정하는 oracle[1]를 정의할 수 있다. 이 Instance를 입력으로 하는 Best-In Greedy을 정의할 수 있다. [2, p. 317–318]

Algorithm 1 Best-In Greedy Algorithm

procedure BEST-IN-GREEDY(A weighted matroid $\mathcal{M} = (E, \mathcal{I})$ with weight function $w : E \to \mathbb{R}_+$)
    Sort $E = \{e_1, ..., e_n\}$ such that $w(e_1) \geq ... \geq w(e_n)$
    Set $T \leftarrow \emptyset$
    for $i \leftarrow 1, n$ **do**
        if $T \cup \{e_i\} \in \mathcal{I}$ **then**
            $T \leftarrow T \cup \{e_i\}$
        end if
    end for
    return T
end procedure

$O(N \log N)$시간 정렬 알고리즘을 사용하면 1의 시간 복잡도는 $O(N \log N)$이다. 따라서 우리는 M차원 각 성분마다 1을 수행하고 크기의 최솟값을 구하면 된다. 이때 independent oracle의 수행 시간을 알아야 한다. Naïve로 하면 independent oracle의 수행 시간은 $O(|T|)$이겠지만 이미 이전에 계산한 값을 다시 이용할 수 있으니 상수 시간에 이를 수행할 수 있다. 따라서 전체 시간 복잡도는 $O(NM \log N)$이다.

참고 문헌

[1] “Oracle machine,” May 2020. [Online]. Available: https://en.wikipedia.org/wiki/Oracle_machine

[2] B. Korte and J. . Vygen, *Combinatorial Optimization Theory and Algorithms*, 4th ed. Springer, 2008.

[1] 임의의 결정 문제 (decision problem)을 하나의 연산으로 해결하는 추상 기계 [1]