Lab04-Matroid

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2021.

$*$ If there is any problem, please contact TA Haolin Zhou.

$*$ Name:Xin Xu Student ID:519021910726 Email: xuxin20010203@sjtu.edu.cn

1. *Property of Matroid.*

   (a) Consider an arbitrary undirected graph $G = (V, E)$. Let us define $M_G = (S, C)$ where $S = E$ and $C = \{I \subseteq E \mid (V, E \backslash I) \text{ is connected}\}$. Prove that M_G is a **matroid**.

   Proof. hereditary. Let $B \in C$, and $A \subset B$. So, $(E \backslash B) \subset (E \backslash A)$. Since $(E \backslash B)$ is connected, $(E \backslash A)$ is connected.
   exchange property. Let $B, A \in C$ and $|A| < |B|$. We will prove exchange property by contradiction. The graph $(E \backslash A)$ must have more than one edges to connect the same two components of graph G. Otherwise, $(E \backslash A)$ only have $v - 1$ edges, and $v =$ the number of vertexes. In this case, $|A|$ is of the largest cardinality among all elements of C, which is contradictory to our hypothesis that $|A| < |B|$. Additionally, B must contain at least one edge e in $(E \backslash A)$ to connect the same two components. If not, $(E \backslash B)$ must hold the same edges of $(E \backslash A)$ except for the single edge connecting two components. But for the single edge, the total number is fixed, which means $|(E \backslash B)|$ is at least the same with$|(E \backslash A)|$ as a result, meaning $|B| \leqslant |A|$ too. So, we just pick this edge e in B, and $A \cup \{e\} \in C$. $\square$

   (b) Given a set A containing n real numbers, and you are allowed to choose k numbers from A. The bigger the sum of the chosen numbers is, the better. What is your algorithm to choose? Prove its correctness using **matroid**.
   Remark: Denote $\mathbf{C}$ be the collection of all subsets of A that contains no more than k elements. Try to prove $(A, \mathbf{C})$ is a matroid.

   Solution. algorithm. Everytime choosing the largest number x in A, and let $A = A \backslash x$. Repeat this process for k times.
   proof. Denote $\mathbf{C}$ be the collection of all subsets of A that contains no more than k elements. And we will try to prove $(A, \mathbf{C})$ is a matroid. **hereditary:** Let $B \in \mathbf{C}$, and $D \subset B$. So D is a set with elements less than k, which is clearly an element of $\mathbf{C}$. **exchange property.** Let $B, D \in \mathbf{C}$ and $|D| < |B|$. For every $x \in B \backslash D, D \cup \{x\}$ is a subset of A, whose cardinality is no more than k. So, $D \cup \{x\} \in \mathbf{C}$. Above all, $(A, \mathbf{C})$ is a matroid, and the greedy algorithm is the optimal method for maximization problem. $\square$

2. *Unit-time Task Scheduling Problem.* Consider the instance of the **Unit-time Task Scheduling Problem** given in class.

   (a) Each penalty ω_i is replaced by $80 - \omega_i$. The modified instance is given in Tab. 1. Give the final schedule and the optimal penalty of the new instance using Greedy-MAX.

Table 1: Task

a_i	1	2	3	4	5	6	7
d_i	4	2	4	3	1	4	6
ω_i	10	20	30	40	50	60	70

   Solution. the final schedule. The order is $a_5, a_4, a_6, a_3, a_7, a_1, a_2$. And the sequence of a_6 and a_3 can be exchanged, which is the same of a_1 and a_2.
   the optimal penalty. The penalty is the sum of a_1 and a_2, which is $w_1 + w_2 = 30$. $\square$

(b) Show how to determine in time $O(|A|)$ whether or not a given set A of tasks is independent. (**Hint**: You can use the lemma of equivalence given in class)

Solution.

Algorithm 1: IndependentSystem

Input: a set A of n tasks.
Output: Whether or not A is an independent system.

```
Sort n elements by penalties so that w_1 ⩾ w_2 ⩾ ... ⩾ w_{n-1} ⩾ w_n;
An array a[n+2]; sum ← 0; j ← 0;
for j ← 0 to n+1 do
    a[j] = 0;
for j ← 1 to n do
    ++a[d_j];
j ← 0;
while sum ⩽ j do
    ++j;
    sum ← sum + a[j];
    if j == n+1 then
        break;
if j < n+1 then
    return false;
return true;
```

$\square$

3. *MAX-3DM.* Let X, Y, Z be three sets. We say two triples (x_1, y_1, z_1) and (x_2, y_2, z_2) in $X \times Y \times Z$ are *disjoint* if $x_1 \neq x_2$, $y_1 \neq y_2$, and $z_1 \neq z_2$. Consider the following problem:

Definition 1 (MAX-3DM)**.** *Given three disjoint sets X, Y, Z and a non-negative weight function $c(\cdot)$ on all triples in $X \times Y \times Z$, **Maximum 3-Dimensional Matching** (MAX-3DM) is to find a collection $\mathcal{F}$ of disjoint triples with maximum total weight.*

(a) Let $D = X \times Y \times Z$. Define independent sets for MAX-3DM.

(b) Write a greedy algorithm based on Greedy-MAX in the form of *pseudo code*.

(c) Give a counter-example to show that your Greedy-MAX algorithm in Q. 3b is not optimal.

(d) Show that: $\max\limits_{F \subseteq D} \frac{v(F)}{u(F)} \leq 3$. (Hint: you may need Theorem 1 for this subquestion.)

Solution. i. Define $\mathbf{C}$ as: $\mathbf{C} = \{F \subseteq D | F \text{ is a collection of disjoint triples}\}$. And prove it an independent system. Let $B \in \mathbf{C}$ and $A \subset B$, it's clear that A is a collection of disjoint triples. So, $A \in \mathbf{C}$.

ii.

Algorithm 2: MAX-3DM

Input: a set D of n triples.

Output: A subset of D with maximum weight.

```
Sort n triples by weights decreasingly so that w_{d_1} ⩾ w_{d_2} ⩾ ... ⩾ w_{d_{n-1}} ⩾ w_{d_n};
A ← ∅;
for j ← 1 to n do
    if A ∪ {d_i} ∈ C then
        A ← A ∪ {d_i};
Output A;
```

iii. The picture of counter example is below. In this case, the greedy algorithm contains a total weight of 11, while the optimal algorithm contains a total weight of 12.

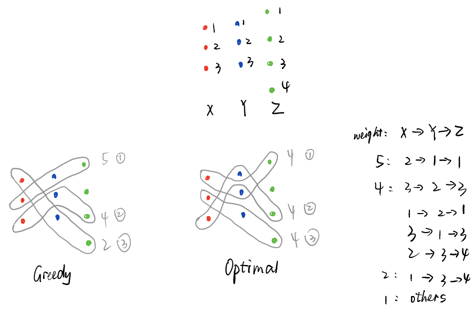

Figure 1: The Counter Example

iv. **Proof.** The independent system $(D, \mathbf{C})$ is the intersection of 3 matroids $(D, \mathcal{C}_i), 1 \leqslant i \leqslant 3$; that is, $\mathbf{C} = \bigcap_{i=1}^{3} \mathcal{C}_i$.
Define $\mathcal{C}_1$ as: $\mathcal{C}_1 = \{F \subseteq D | F$ is a collection of triples that any $x_i \neq x_j$ if $i \neq j.\}$.
Define $\mathcal{C}_2$ as: $\mathcal{C}_2 = \{F \subseteq D | F$ is a collection of triples that any $y_i \neq y_j$ if $i \neq j.\}$.
Define $\mathcal{C}_3$ as: $\mathcal{C}_3 = \{F \subseteq D | F$ is a collection of triples that any $z_i \neq z_j$ if $i \neq j.\}$.
And it's easy to prove that $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$ are all matroids. According to **Theorem 1.**, $\max\limits_{F \subseteq D} \frac{v(F)}{u(F)} \leq 3$. □

□

Theorem 1. *Suppose an independent system $(E, \mathcal{I})$ is the intersection of k matroids $(E, \mathcal{I}_i)$, $1 \leq i \leq k$; that is, $\mathcal{I} = \bigcap_{i=1}^{k} \mathcal{I}_i$. Then $\max\limits_{F \subseteq E} \frac{v(F)}{u(F)} \leq k$, where $v(F)$ is the maximum size of independent subset in F and $u(F)$ is the minimum size of maximal independent subset in F.*

Remark: You need to include your .pdf and .tex files in your uploaded .rar or .zip file.