CSL 356 Lecture 15 Sept 3

# Proof of the matroid theorem

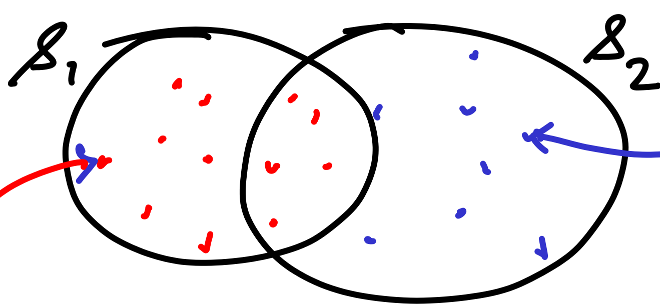
① ⇒ ②

Suppose greedy solves the problem optimally for a subset system $(S, M)$ for any weight function.

We want to show that the exchange property holds.

Suppose not (proof by contradiction)

Then there exists $S_1$, $S_2$ $|S_1| = i$ $|S_2| = i+1$ for which exchange property doesn't hold. Let define the weight function as




weight $= i+2$

weight $= i+1$

For rest of elements i.e. $S - S_1 - S_2$, weight $= 0$

basic greedy will pick all elements of $S_1$ (before $S_2 - S_1$) and then it gets stuck

greedy solution is $(i+2)\,i$

$$= i^2 + 2i$$

On the other hand, if we had picked all elements of $S_2$, the cost is at least $(i+1)^2 > i^2 + 2i$

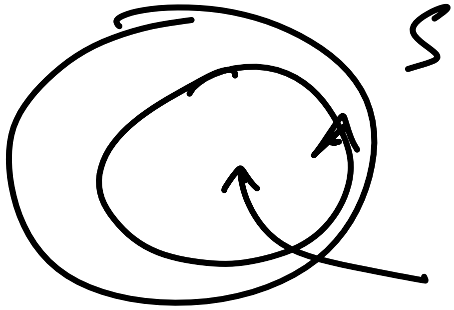
i.e. greedy doesn't work

contradiction

$(2) \Rightarrow (3)$

exchange property

(3): all maximal subsets have same size




maximal subsets within A

$S_1^A \quad S_2^A \quad \ldots$ independent

if $S_1^A$ and $S_2^A$ are maximal then they must have the same size as we can add an element $x \in S_2^A - S_1^A$ to $S_1^A$ by ex property

③ ⇒ ① : basic greedy solves the problem optimally

all maximal subsets have same size

Proof (by contradiction).

Suppose greedy doesn't solve the problem optimally for some instance. Let us enumerate the subset produced by greedy in decreasing order of weights

greedy output $\tilde{x}_1 \geq \tilde{x}_2 \geq \tilde{x}_3 \ldots \boxed{\tilde{x}_i} \ldots \tilde{x}_l$

optimal soln $\tilde{y}_1 \geq \tilde{y}_2 \geq \tilde{y}_3 \ldots \boxed{\tilde{y}_i} \ldots \tilde{y}_l$

→ decreasing wts

$w(\tilde{y}_i) > w(\tilde{x}_i)$ : let $i$ be the first such element.

(otherwise greedy is better than opt)

$A \subset S$ s.t. $A = \{ x \in S \mid w(x) \geq w(\tilde{y}_i) \}$

Observ. $\tilde{x}_1, \tilde{x}_2 \ldots \tilde{x}_{i-1} \in A$ maximal in $A$

# Scheduling Problem

Given a set of jobs $J_1, J_2 \ldots J_n$ every $J_i$ has a deadline $d_i$ associated with it and if $J_i$ cannot complete within $d_i$, then we incur a penalty. Further each $J_i$ takes unit time.

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| deadlines | 3 | 1 | 2 | 2 |
| Penalty | 5 | 3 | 6 | 10 |

$J_2 \to J_4 \to J_1$

$J_4 \to J_2 \to J_1$ ✗

penalty: 6

Objective function: minimize the penalty of jobs not scheduled

$\Rightarrow$ maximize the penalty of jobs scheduled

__Claim__

The Job scheduling problem satisfies exchange property