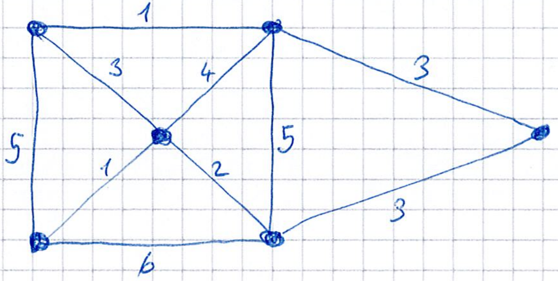
# Matroids and optimization

We consider, as a first example of an optimization problem, the maximum/minimum weight spanning tree problem.

Given a weighted graph $G = (V, E, a, b)$ with weight function $w: E \to \mathbb{R}$, say that the weight of any $A \subseteq E$ is

$$w(A) = \sum_{a \in A} w(a) \qquad (\text{"linear weight function"})$$

As an example, take the following graph.



Say, we want to find a spanning tree $T \subseteq E$ with minimal weight. Two of the most renowned algorithms are as follows

Prim's algorithm: choose a starting vertex ("root") and, from there, "grow" a tree choosing at each step the "lightest" edge.

Kruskal's algorithm: Start with an empty edge-set and add at ~~every~~ step ~~the~~ "lightest" edge that does not generate a cycle.

Both these algorithms are "greedy", in the sense that each step goes for "local optimality"/"immediate win". One has, then, to prove that such algorithms are correct - that is: they do output an optimal tree. In a class on algorithms/complexity this can be done in an elementary way. Here we strive for a structural characterization.

First, let us make precise what we mean by GREEDY.

Both algorithms above "explore" some set of so-called <u>feasible</u> configurations:

- "Trees rooted at $v$" in Prim's case
- "All forests" in Kruskál's case

These configurations can be taken as subsets of a ground set $E$ and can be "reached" from an "empty" configuration by means of elementary steps. More precisely, the family $\mathcal{F} \subseteq 2^E$ of feasible configurations satisfies

(H) For every $X \in \mathcal{F}$ there is $x \in X$ with $X \setminus \{x\} \in \mathcal{F}$.

Notice that (H) and finiteness of $E$ imply $\emptyset \in \mathcal{F}$.

Moreover, a weight function $w: \mathcal{F} \to \mathbb{R}$ is given.

The optimization problem associated to $\mathcal{F}$ and $w$ is then to find a maximal (by inclusion) member of $\mathcal{F}$ whose weight is minimal possible. This means finding $X \in \max_{\subseteq} \mathcal{F}$ such that $w(X) = \min\{w(A) \mid A \in \max_{\subseteq} \mathcal{F}\}$.

The "Greedy" algorithm then is formalized as follows.

```
GREEDY:
 Set $X := \emptyset$
WHILE $X \notin \max_{\subseteq} \mathcal{F}$
   DO choose $e \in E \setminus X$ s.t.  $X \cup \{e\} \in \mathcal{F}$,
                                         $w(X \cup \{e\}) = \min \left\{ w(A) \;\middle|\; \begin{matrix} A \in \mathcal{F} \\ |A \setminus X| = 1 \end{matrix} \right\}$.
        set $X \leftarrow X \cup \{e\}$.

OUTPUT  $X$.
```

In our instances, the weight function is "linear" in the sense that it is induced by an <u>elementary weight function</u> $w : E \to R$ via $w(A) = \sum_{a \in A} w(a)$.

Even more specially, in the case of Kruskal's algorithm, the family $\mathcal{F}$ not only satisfies (M), but it even satisfies (I1), i.e., it is an abstract simplicial complex.

The proof of correctness of Kruskal's algorithm now follows from the following general theorem

<u>Theorem</u> An abstract simplicial complex $\mathcal{F} \subseteq 2^E$ is the set of independent sets of a matroid if and only if,

$$(S)\left\{\begin{array}{l}\text{for every linear weight function } w : \mathcal{F} \to \mathbb{R}, \text{ the optimization} \\ \text{problem associated to the induced weight function on } \mathcal{F} \\ \text{is solved correctly by the greedy algorithm}\end{array}\right.$$

<u>Lemma 1</u> Let $\mathcal{I}$ be the set of independent sets of a matroid on $E$ and let $w: E \to \mathbb{R}$ be any elementary weight function. Then, the optimization problem on $\mathcal{I}$ with linear weight function induced by $w$ is solved by the GREEDY algorithm.

<u>Proof</u>: The greedy algorithm terminates with a basis, say $B_G = \{e_1, \dots, e_r\}$, where the indexing reflects the ordering of "choice" of the $e_i$. Then, clearly

$$w(e_1) \leq \dots \leq w(e_r).$$

Let now $B = \{f_1, \dots, f_r\}$ be any other basis of the matroid, indexed so that $w(f_1) \leq \dots \leq w(f_r)$.

<u>Claim</u> For all $1 \leq j \leq r$, $w(e_j) \leq w(f_j)$

<u>Pf</u>: Assume BWOC that the claim fails, and let $k$ be the smallest index with $w(e_k) > w(f_k)$.
Then $I_1 := \{e_1, \dots, e_{k-1}\}$, $I_2 := \{f_1, \dots, f_k\}$ are independent sets with $|I_2| > |I_1|$, so with (I3) there is $1 \leq t \leq k$ with $I_1 \cup \{f_t\}$ independent, $f_t \in I_2 \setminus I_1$
<u>But</u>: $w(f_t) \leq w(f_k) < w(e_k)$, hence GREEDY should have chosen $f_t$ instead of $e_k$! ↯ $\lrcorner$

The claim implies in particular that $B_G$ is optimal

$\boxtimes$

Lemma 2 Let $\mathcal{F} \subseteq 2^E$ satisfy (I1), (I2), (S). Then $\mathcal{F}$ satisfies (I3).

Proof BWOC Let $I_1, I_2 \in \mathcal{F}$ be such that

$$|I_1| < |I_2| \quad \text{but} \quad I_1 \cup \{e\} \notin \mathcal{F} \quad \text{for all } e \in I_2 \setminus I_1 .$$

we want to find a weight function whose optimization problem is not solved by GREEDY. Consider the elementary weight function

$$w: E \to \mathbb{R}, \quad e \mapsto \begin{cases} -1 & \text{if } e \in I_1 \\ -\varepsilon & \text{if } e \in I_2 \setminus I_1 \\ 0 & \text{else} \end{cases}$$

where $\varepsilon$ is any number $0 < \frac{|I_1 \setminus I_2|}{|I_2 \setminus I_1|} < \varepsilon < 1$

The greedy algorithm will then pick all elements of $I_1$ first, and then some elements of $I_2^c$, since $I_1 \cup \{e\} \notin \mathcal{F}$ for $e \in I_2 \setminus I_1$. Hence $w(B_G) = -|I_1|$

However, if we pick any $B \in \max \mathcal{F}$, $B \supseteq I_2$, we see

$$\begin{array}{rcccl} w(B) & = & w(B \cap I_1) & + & w(B \setminus I_1) \\ & & \rotatebox{90}{$\leq$} & & \parallel \; w(f) = 0 \;\forall f \notin I_2 \\ & \leq & w(I_2 \cap I_1) & + & w(I_2 \setminus I_1) \\ & & \parallel & & \\ & & -|I_2 \cap I_1| & - & \varepsilon |I_2 \setminus I_1| \\ & & & & \\ & < & -|I_2 \cap I_1| & - & |I_1 \setminus I_2| \quad = -|I_1| = w(B_G) \end{array}$$

therefore $B_G$ cannot be optimal.

Proof of the theorem follows combining the two Steps. ☒

At this point we still cannot say anything on the correctness of Prim's algorithm. In fact, in this case $\mathcal{F}$ is not a simplicial complex, hence $\mathcal{F}$ cannot define a matroid. Still, we can make the following definition:

Definition Any $\mathcal{F} \subseteq 2^E$ satisfying (M) and (I3) is the set of feasible sets of a greedoid on $E$.

Remark: Since (I1) & (I2) => (M), every matroid is a greedoid. But there are greedoids that are not matroids (see set of feasible sets of Prim's algorithm).

In order to state the main theorem about optimization on greedoids, let us talk "languages".

Let $E$ be a finite set.

Then, let $E^*$ denote the set (free monoid) of all *words* over the alphabet $E$. Informally, any $\alpha \in E^*$ is of the form $\alpha = x_1 x_2 x_3 \dots x_k$ for some $k \in \mathbb{N}$ and some $x_i \in E$.

We write $\alpha\beta$ for the *concatenation* of any two $\alpha, \beta \in E^*$.

For any word $\alpha \in E^*$, let

$|\alpha|$ denote the *length* of $\alpha$, i.e., the number of (not necessarily distinct) letters in $\alpha$

$\tilde{\alpha}$ denote the *support* of $\alpha$, i.e., the *set* $\tilde{\alpha} \subseteq E$ of all letters that appear in $\alpha$.

A word $\alpha \in E^*$ is called *simple* if $|\alpha| = |\tilde{\alpha}|$, i.e., every letter appears at most once.

*Example*: $E = \{x, y, z\}$ $\alpha = xxyzxz$ not simple:

$|\alpha| = 6$, $\tilde{\alpha} = \{x, y, z\}$

A *language* over $E$ is any $\Lambda \subseteq E^*$. A language is called *simple* if all of its elements are simple words.

If $\Lambda$ is a language, write $\widetilde{\Lambda} := \{\tilde{\alpha} \mid \alpha \in \Lambda\}$

<u>Definition</u> A <u>greedoid language</u> over a finite ground set $E$ is a simple language $\Lambda \subseteq E^*$ such that

(L1) If $\alpha = \beta\gamma$ and $\alpha \in \Lambda$, then $\beta \in \Lambda$

("left hereditarity")

(L2) If $\alpha, \beta \in \Lambda$ and $|\alpha| > |\beta|$ then $\alpha$ contains a letter $x$ s.t. $\beta x \in \Lambda$

<u>Proposition</u> / <u>remark</u> (Equivalence greedoids $\leftrightarrow$ greedoid languages)

(i) If $\Lambda$ is a greedoid language over $E$, then $\tilde{\Lambda}$ is the set of feasible sets of a greedoid over $E$.

(ii) If $\mathcal{F}$ is the set of feasible sets of a greedoid over $E$, then
$$\Lambda(\mathcal{F}) := \{x_1 \dots x_k \in E^* \mid \{x_1, \dots, x_j\} \in \mathcal{F} \ \ \forall j \in \{1 \dots k\}\}$$
is a greedoid language.

(iii) Moreover, $\Lambda(\tilde{\Lambda}) = \Lambda$ and $\widetilde{\Lambda(\mathcal{F})} = \mathcal{F}$.

<u>Example</u>: some words from the greedoid language of the greedoid of rooted trees in Example 1.

A maximal (i.e., non-extendable) word in a simple language is called a <u>basic word</u>.

The formulation of greedoids in terms of languages suggests considering weight functions of the form

$$w : \Lambda \longrightarrow \mathbb{R}$$

and optimization problems of the form:

"find a basic word $x \in \Lambda$ that minimizes $w$"

Again, this task can be tackled by a greedy algorithm

```
GREEDY
Set $\alpha$ the empty word
WHILE $\alpha$ not basic
  DO choose $x \in E$ s.t. $\begin{cases} \alpha x \in \Lambda , \\ w(\alpha x) \leq w(\alpha y) \text{ for all } y \in E \text{ s.t. } \alpha y \in \Lambda . \end{cases}$
     set $\alpha \leftarrow \alpha x$
OUTPUT $\alpha$.
```

Clearly if $w(\alpha)$ only depends on $\tilde{\alpha}$ we recover the previous version of the algorithm.

<u>Definition</u> A weight function $w : \Lambda \to \mathbb{R}$ is "compatible" if, whenever $\alpha x \in \Lambda$ satisfies $w(\alpha x) \leq w(\alpha y)$ for all $y \in E$, $\alpha y \in \Lambda$,

- $\alpha\beta x\gamma \in \Lambda$ and $\alpha\beta z\gamma \in \Lambda$ imply $w(\alpha\beta x\gamma) \leq w(\alpha\beta z\gamma)$
- $\alpha x\beta z\gamma \in \Lambda$, $\alpha z\beta x\gamma \in \Lambda$ imply $w(\alpha x\beta z\gamma) \leq w(\alpha z\beta x\gamma)$

(Notice: these conditions state some kind of "global homogeneity")

Remark: If $w(\alpha)$ only depends on $w(\tilde{\alpha})$, the second condition is trivially satisfied.

Moreover, if $w(\alpha)$ is the value on $\tilde{\alpha}$ of some linear weight function, then the first condition is satisfied as well.

Theorem: Let $\Lambda$ be a simple language on $E$ satisfying (L1). Then $\Lambda$ is a greedoid language if and only if the greedy algorithm gives an optimal solution for every compatible weight function.

Corollary: Prim's algorithm is correct!

# Duals of representable matroids

**Theorem** If a matroid $M$ is representable over some field $\mathbb{F}$, then so is $M^*$.

**Proof** Wlog suppose that $M$ is represented by a $d \times n$-matrix

$$A := \left[\begin{array}{c|c} I_d & N \end{array}\right]$$

where $N$ is a $d\times(n-d)$ matrix, and $d = rk(M)$.

We claim that $M^*$ is represented by

$$A^* := \left[\begin{array}{c|c} -N^T & I_{(n-d)\times(n-d)} \end{array}\right]. \quad \text{an } (n-d)\times n \text{-matrix}$$

To prove this pick a basis of $M$ and (rearranging colums and rows) suppose

$$A := \left[\begin{array}{c|c|c|c} I & \overbrace{O & N_{11}}^{B} & N_{12} \\ \hline O & \underbrace{I & N_{21}} & N_{22} \end{array}\right] \begin{array}{l} \}t \\ \}d-t \end{array}$$

Then

$$A^* := \left[\begin{array}{c|c|c|c} -N_{11}^T & -N_{21}^T & I & O \\ \hline -N_{12}^T & -N_{22}^T & O & \underbrace{I}_{n-d-t} \end{array}\right] \begin{array}{l} \}t \\ \}n-d-t \end{array}$$

Now:

- Since $\begin{array}{c|c} O & N_{11} \\ \hline I & N_{21} \end{array}$ has full rank, then $N_{11}$ (and $-N_{11}^T$) have rank $t$.
- thus $\begin{array}{c|c} N_{11}^T & O \\ \hline N_{12}^T & I \end{array}$ has full rank,

hence rank $n-d$. but the rank of $A^*$ can't be more than $n-d$, hence that is a basis. $\boxtimes$