Stockholm University

Department of Mathematics

Mathematics III: Combinatorics

Theoretical Compendium

Sofia Tirabassi

Lecture Notes

2025

Abstract

The textbook for the course MM5023- Mathematics III Combinatorics, given at the department of Mathematics of Stockholm University has been for more than a decade Grimaldi's *Discrete and Combinatorial Mathematics an Applied Introduction.* The textbook is rich of examples and provide a good collection of exercises, many of which are the same level of the exam. However, it lacks the Definition-Statement-Proof structure typical of mathematics literature. No other textbook is as wide in scope as Grimaldi's, nor presents so many worked out examples, and their exercises are, in my opinion, lacking. The theoretical ones are too remote and difficult, the computational ones are too easy. With these lecture notes, I aim at integrating Grimaldi's book with some theoretical foundations and give more structure to the material covered in the lectures.

Contents

List of Algorithms

Baseline Theorems

Here is a list of results that you should be able to prove.

- Pigeonhole principle 1.4.1
- Generalized pigeonhole principle 1.4.8
- Calculation of the Euler ϕ-function 2.3.1
- Multiplicativity of rook polynomials 3.1.4
- Recursive formula to compute rook polynomials 3.1.5
- Generating function of partitions of natural numbers 4.3.6
- The summation operation 5.2.1
- Existence and unicity of the solution of the first order linear recurrence relation with boundary conditions 6.1.2
- Every walk of minimal length is a path 8.1.8
- Degree formula 9.1.2
- Any finite connected graph has a spanning tree 11.1.5
- The value of a flow can be calculated on every cut 13.1.5
- Being parallel is an equivalence relation 14.1.5

List of Definitions

Notation

We will denote the standard number sets by the usual letter in blackboard bold:

$$\mathbb{N},\ \mathbb{Z},\ \mathbb{Q},\ \mathbb{R}.$$

We follow the convention that 0 is a natural number. We are also going to use the standard decorations. For example

$$\mathbb{R}^+ := \{x \in \mathbb{R} \,|x > 0\}$$

$$\mathbb{R}^- := \{x \in \mathbb{R} \,|x < 0\}$$

$$\mathbb{R}^* := \{x \in \mathbb{R} \,|x \neq 0\}$$

$$\mathbb{R}_{\geq 0} := \{x \in \mathbb{R} \,|x \geq 0\}$$

The symbol $:=$ appearing above means **equal by definition**.

Given a set X we have that $\mathscr{P}(X)$ denotes the **power set** of X, that is the family of all the subsets of X.

Quantifiers and other logic symbols

Sometimes we will use the following notation from logic

- $\forall$ means "for all";
- $\exists$ means "exists";
- $\nexists$ means "it does not exist";
- $\Rightarrow$ means "implies";
- $\Leftrightarrow$ means "equivalence"

LECTURE 1

Review

Lecture Plan

1.1 Review of (naive) set theory

A **set** is a well-defined collection of objects, which we call elements. To say that an object x belongs to a set A, we write

$$x \in A.$$

If otherwise x is not an element of A then we write $x \notin A$.

Example 1.1.1. There are many ways to describe a given set. We can (especially if the set is finite) list its element, describe them with words or via a mathematical law.

$$A = \{1, 2, 3\} = \{\text{Integers between 1 and 3 included}\} = \{x \in \mathbb{Z} \mid 0 < x < 4\}$$

We denote the **empty set**, the only set with no elements, by $\emptyset$.

Given two sets A and B, we say that A is **equal** to B if and only if they have the same elements. We say that A is a **subset** of B, and we write $A \subseteq B$ if every element of A is also an element of B. We say that A is a **proper subset** of B - and we write $A \subsetneq B$ or $A \subset B$ - if $A \subseteq B$ and there is an element $b \in B$ such that $b \notin A$.

Example 1.1.2. Observe that the empty set is a subset of every set. In fact, suppose that this is not the case and there is a set A such that $\emptyset \not\subseteq A$. This means that there is an element of the empty set that is not in A, contradicting the definition of the empty set as the set with no elements.

Proposition 1.1.3. *We have that $A = B$ if and only if $A \subseteq B$ and $B \subseteq A$.*

Proof. Omitted. ∎

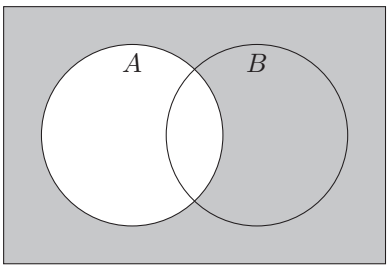


Figure 1.1: Complementary of A

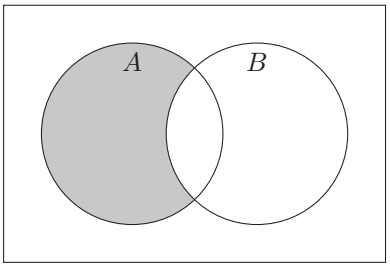


Figure 1.2: Excision

Example 1.1.4. Attention about sets that are elements of a set! If we have the following set

$$A := \{\{1, 2\}, 3, 4, 5, \{6\}\}$$

We have that $\{1, 2\} \in A$ and $6 \notin A$. In the same way, $\{6\} \not\subset A$, but $\{6\} \in A$ and $\{\{6\}\} \subseteq A$

In order to leave barbers (and associated paradoxes) shaving themselves without any trouble, we will assume that all the sets that we encounter will be subsets of a given subset S, which will be our universe. Given a set S and A a subset of S, the **complementary of** A is

$$A^c := \{x \in S \mid x \notin A\}$$

Example 1.1.5.

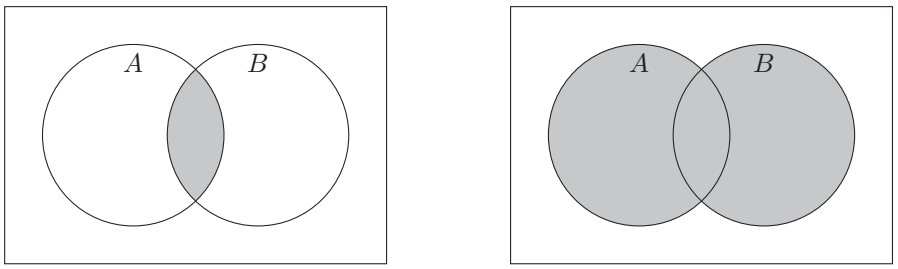
- If $A = S$ then $A^c = \emptyset$.
- If $A = \emptyset$, then $A^c = S$
- If $S = \mathbb{R}$ and $A = (0, 1]$ then $A^c = (-\infty, 0] \cup [1, \infty)$.

Given two sets A and B we have that the **excision** of A by B is

$$A \backslash B := \{x \in A \mid x \notin B\}$$

Let S be a set (it will be our Universe). Given a (possible infinite) set of indices $\mathcal{I}$, for every $i \in I$ we consider a subset A_i of S. Then we define

$$\bigcup_{i \in I} A_i := \{x \in S \mid \exists\, i \in \mathcal{I} \text{ such that } x \in A_i\}$$



(a) Intersection (b) Union

Figure 1.3: Intersection and Union

$$\bigcap_{i\in I} A_i := \{x \in S \mid \forall\, i \in \mathcal{I},\ x \in A_i\}$$

Example 1.1.6. Let $S = \mathbb{R}$ and $\mathcal{I} = \mathbb{N}$ (during this class we will work under the convention that 0 is not a natural number).

- $\bigcup_{n\in\mathbb{N}}[-n, n] = \mathbb{R}$
- $\bigcap_{n\in\mathbb{N}}[-n, n] = [-1, 1]$
- $\bigcup_{n\in\mathbb{N}}\left[0, \frac{1}{n}\right] = [0, 1]$
- $\bigcap_{n\in\mathbb{N}}\left[0, \frac{1}{n}\right] = \{0\}$

Theorem 1.1.7 (De-Morgan)**.** *Given a set of indices $\mathcal{I}$, a set S and a collection of subsets of S indexed by $\mathcal{I}$, $\{A_i\}_{i\in\mathcal{I}}$, then the following statements are true.*

1. $\left(\bigcup_{i\in\mathcal{I}} A_i\right)^c = \bigcap_{i\in\mathcal{I}} A_i^c$
2. $\left(\bigcap_{i\in\mathcal{I}} A_i\right)^c = \bigcup_{i\in\mathcal{I}} A_i^c$

Proof. We are just going to prove the first assertion and we leave the second as an exercise.

Let $x \in \left(\bigcup_{i\in\mathcal{I}} A_i\right)^c$, this means by definition that (is equivalent to) $x \notin \bigcup_{i\in\mathcal{I}} A_i$. So it is not true that there is an $i \in \mathcal{I}$ such that $x \in A_i$. Equivalently for every $i \in \mathcal{I}$, $x \notin A_i$. But by definition of intersection we have that this is in turn equivalent to $x \in \bigcap_{i\in\mathcal{I}} A_i^c$. ■

Theorem 1.1.8 (Distributivity)**.** *Given a set of indices $\mathcal{I}$, a set S and a collection of subsets of S, $\{A_i\}_{i\in\mathcal{I}}$, and another subset of S, B, then the following statements are true.*

1. $B \cap \left(\bigcup_{i\in\mathcal{I}} A_i\right) = \bigcup_{i\in\mathcal{I}}(A_i \cap B)$
2. $B \cup \left(\bigcap_{i\in\mathcal{I}} A_i\right) = \bigcap_{i\in\mathcal{I}}(A_i \cup B)$

Proof. We show the first statement and leave the second as an exercise.

Suppose first that $x \in B \cap \left(\bigcup_{i\in\mathcal{I}} A_i\right)$. By definition this is equivalent to $x \in B$ and $x \in \bigcup_{i\in\mathcal{I}} A_i$. Again, this is equivalent to x being an element of B and to the fact that there is some $j \in \mathcal{I}$ such that $x \in A_j \cap B$. In particular we deduce that $x \in \bigcup_{i\in\mathcal{I}}(A_i \cap B)$ and that $B \cap \left(\bigcup_{i\in\mathcal{I}} A_i\right) \subset \bigcup_{i\in\mathcal{I}}(A_i \cap B)$
Suppose conversely that $x \in \bigcup_{i\in\mathcal{I}}(A_i \cap B)$, then there is an $j \in \mathcal{I}$ such that $x \in A_j \cap B$. Therefore $x \in B$ and $x \in A_j$. In particular $x \in B$ and $x \in \bigcup_{i\in\mathcal{I}} A_i$. We deduce that $x \in B \cap \left(\bigcup_{i\in\mathcal{I}} A_i\right)$, and therefore $B \cap \left(\bigcup_{i\in\mathcal{I}} A_i\right) \supset \bigcup_{i\in\mathcal{I}}(A_i \cap B)$. Then equality of the two sets is proven. ■

Definition 1.1.9 (Disjoint Union)**.** We say that a set C is the **disjoint union** of two sets A and B - and we write $C = A \sqcup B$ - if $C = A \cup B$ and $A \cap B = \emptyset$.

Example 1.1.10. The set of integers $\mathbb{Z}$ is the disjoint union of the set of even integers and the set of odd integers.

1.2 Review of functions

Given two sets A and B, a function $f : A \to B$ is an assignment that to **each** element a of A associate a **unique** element b of B. We write $b = f(a)$, or $a \mapsto b$. The set A is called the **domain of** f, while B is called the **codomain** of f. Two functions $f : A \to B$ and $g : A_1 \to B_1$ are equal if and only if they have same domain ($A = A_1$), same codomain ($B = B_1$) and, for every $a \in A$, $f(a) = g(a)$ (same law).

Example 1.2.1.

- The assignment from $\mathbb{R} \to \mathbb{R}^+$ that send x to x^2 is not a function, since there is no $f(0)$.
- Let $A = \{1, 2, 3\}$ and $B = \{\{1, 2\}, \{2, 3\}\}$, the assignment $f : A \to B$ such that $f(a) = b$ if and only if $a \in b$ is not a function, since there is not just a unique output for 2.
- $f : \mathbb{R} \to \mathbb{R}$ defined by $x \mapsto x^2$ ($f(x) = x^2$) is not equal to $g : \mathbb{R} \to \mathbb{R}_{\geq 0}$ defined by $g(x) = x^2$, since they do not have the same codomain.

Given a function $f : A \to B$, the **range** or **image** of f is

$$f(A) := \{b \in B \mid \exists\, a \in A \text{ such that } b = f(a)\} \subset B.$$

More generally, if A_1 is a subset of A, the **image of** A_1 **under** f is

$$f(A_1) := \{b \in B \mid \exists\, a \in A_1 \text{ such that } b = f(a)\} \subset B.$$

If B_1 is a subset of B the **preimage** of B_1 under f is

$$f^{-1}(B_1) = \{a \in A \mid f(a) \in B_1\} \subset A.$$

If $B = \{b\}$ is a singleton (a set with just one element) then we write $f^{-1}(\{b\})$ as simply $f^{-1}(b)$. Attention: this is not to be mixed with the image of b through the inverse function of f (see later).

Example 1.2.2. Let $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = x^2$, then $f([-2, 1]) = [0, 2]$, $f^{-1}(-1) = \emptyset$, $f^{-1}(4) = \{2, -2\}$.

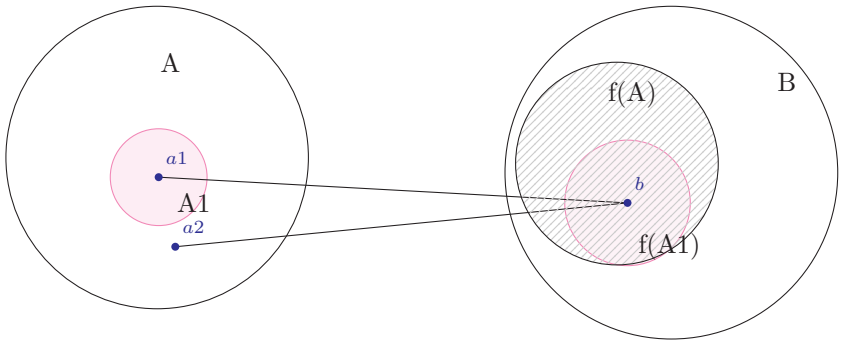


Figure 1.4: Graphical visualization of a function

We say that a function $f : A \to B$ is **injective** (one-to-one) if $f(x) = f(y)$ implies that $x = y$. Observe that this is equivalent to the fact that for every $b \in B$, $f^{-1}(b)$ has at most one element. We say that it is **surjective** (onto) if $f(A) = B$ (this is equivalent to $f^{-1}(b)$ being nonempty for every $b \in B$.). A function that is both injective and surjective is said to be **bijective** (a one-to-one correspondence). In this case we can define the **inverse** of f as the function $f^{-1} : B \to A$ such that $f^{-1}(b) = a$ if and only if $f(a) = b$. Observe that this is indeed a function. In fact, since f is surjective, every b in B can be written as $b = f(a)$ and so every b admit an assignment through f^{-1} in addition this assignment is unique since the injectivity of f grants that there is just one $a \in A$ such that $f(a) = b$.

Example 1.2.3.

- Let A be a set, $\mathrm{id}_A : A \to A$ is the function defined by $a \mapsto a$. It is called the **identity function of** A. It is both injective and surjective.

- Let B be a set and $A \subset B$. We can define a function $i_A : A \to B$ by the assignment $a \mapsto a$. This is the **canonical immersion** of A into B. It is injective but not surjective unless $A = B$ (in that case $i_A = \mathrm{id}_A$).

If $f : A \to B$ is a function and $g : B \to C$ is another function then we can define the **composition of** f **and** g as the function $g \circ f : A \to B$ which to each $a \in A$ assigns the element $g(f(a))$ in C, that is $g \circ f(a) = g(f(a))$.

1.3 Functions and Relations (that is functions done right)

The Cartesian product of two sets A and B is

$$A \times B := \{(a, b) \mid a \in A \text{ and } b \in B\}$$

Definition 1.3.1 (Relation)**.** A **relation** $\mathfrak{R}$ from a set A to a set B is a subset of $A \times B$. If $(x, y) \in \mathfrak{R}$ we write $x\mathfrak{R}y$ and we say that x is in relation with y.

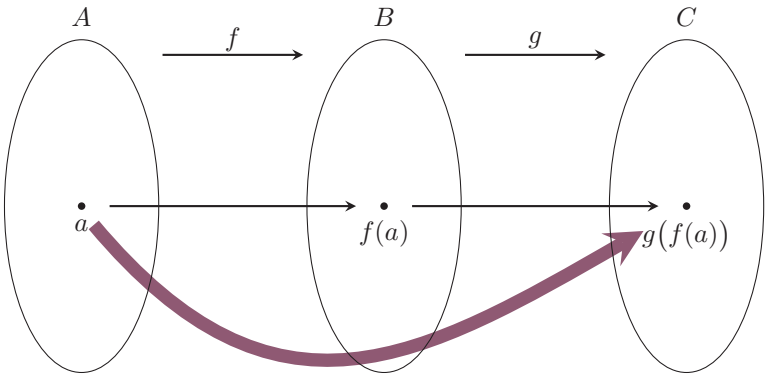


Figure 1.5: Composition of functions

When $A = B$ we say that a relation $\mathfrak{R}$ is **reflexive** if $x\mathfrak{R}x$ for every $x \in A$. The relation is **symmetric** if $y\mathfrak{R}x$ whenever $x\mathfrak{R}y$. It is **transitive** if $x\mathfrak{R}z$ whenever $x\mathfrak{R}y$ and $y\mathfrak{R}z$.

Using relations we can give a more formal definition of functions:

Definition 1.3.2 (Formal definition of functions)**.** A function $f : A \to B$ is a relation $\Gamma \subseteq A \times B$ such that

For every $x \in A$, there is a unique $y \in B$, denoted by $f(x)$ such that $x\Gamma y$.

Remark 1.3.3. We are basically identifying a function with its graph: let $f : A \to B$ as in Section 1.2. The graph of f is

$$\Gamma_f \coloneqq \{(x, y) \in A \times B \mid y = f(x)\} \subseteq A \times B.$$

Then Γ_f is a relation from A to B satisfying the condition of Definition 1.3.2.

1.4 Counting with functions

Let n be a positive integer, we introduce the following set:

$$\underline{n} \coloneqq \{1, 2, \ldots, n\}.$$

The main idea behind counting with functions is that we want to say that a set has n elements if and only if there is a bijection between it and $\underline{n}$. For this to be well-defined we have to ensure that there are no bijections between $\underline{n}$ and $\underline{m}$, when $n \neq m$. This is ensured by the pigeonhole principle:

Principle 1.4.1. *(Pigeonhole principle) There is an injective function $f : \underline{n} \to \underline{m}$ if, and only if $m \geq n$.*

Remark 1.4.2. We call this a principle, because its analogue for sets of infinite cardinality/size cannot be proven, but has to be taken as an axiom.

Proof. This is a consequence of the well-ordering of the natural numbers. That is that every non-empty subset of the natural numbers has a minimum. Suppose, by contradiction that the principle fails. Then the following set is non-empty:

$$S := \{m \in \mathbb{N} \mid \exists\, n \in \mathbb{N},\ n > m \text{ and } f : \underline{n} \to \underline{m} \text{ injective}\}.$$

Denote by d the minimum of S and take an $n > d$ and an injective map $f : \underline{n} \to \underline{d}$. Up to permuting the elements of $\underline{d}$ we can assume that $n \mapsto d$. Then we can consider $g : \underline{n-1} \to \underline{d-1}$, obtained by restricting f to $\underline{n-1}$. Clearly, g is injective, and $m - 1 > d - 1$, contradicting the minimality of d. ■

Exercise 1.4.3 (Challenging exercise)**.** There is a team chess tournament. There are 5 teams of 2 players each. The rules are

- No one can play a game against their partner;
- No one can play a game against the same opponent twice.

At the end of the tournament, Aragorn asks each person how many games they have played, and he gets different answers. How many games has Bilbo (Aragorn's teammate) played?

Now we can define the following.

Definition 1.4.4 (Finite set)**.** We say that a set A is **finite** if there is a natural number n and a bijective function $f : A \to \underline{n}$. If this is the case then we say that A has **size** (cardinality) n, and we write $|A| = n$.

Remark 1.4.5. Is this is a correct definition? Can it happen that A admits bijections to $\underline{n}$ and $\underline{m}$ for distinct n and$m$? The answers to these questions are YES and NO, thanks to the pigeonhole principle. In fact, suppose that $m > n$. Let $f : A \to \underline{n}$ and $g : A \to \underline{m}$ be bijections. Then $g^{-1} \circ f : \underline{m} \to \underline{n}$ would be a bijection. In particular we would have an injective map $\underline{m} \to \underline{n}$, contradicting the pigeonhole principle.

Remark 1.4.6. The usual pigeonhole principle is stated as follows

> If there are $n + 1$ pigeons and n nests then there is at least a nest with two pigeons.

This is a direct consequence of the Principle stated above. Let N be the set of nests and P be the set of pigeons. Now let $f : P \to N$ be the function that assign to each pigeon its nest. By the pigeonhole principle (Principle 1.4.1), f cannot be injective. In particular there is a nest y such that $f^{-1}(y)$ has size at least 2.

Formal rule of sum

Proposition 1.4.7.

$$|A \sqcup B| = |A| + |B|$$

Proof. Let $|A| = n$ and $|B| = m$, so that there are bijective functions $f_A : A \to \underline{n}$ and $f_B : B \to \underline{m}$. Let us define a function $f : A \sqcup B \to \underline{m+n}$ via the assignment

$$f(x) = \begin{cases} f_A(x) & \text{if } x \in A \\ n + f_B(x) & \text{if } x \in B \end{cases}$$

This is a well-defined function because $A \cap B = \emptyset$. This is surjective because both f_A and f_B are, and it is injective because both f_A and f_B are so. ■

The rule of sum as stated in the book says

> If a task can be performed in m ways and a second task can be performed in n ways, then, provided that the two tasks cannot be performed simultaneously we have $m + n$ choices of actions to perform.

One can translate this in the formal rule of sum by setting

$$A \coloneqq \{\text{ways to perform task 1}\}$$

$$B \coloneqq \{\text{ways to perform task 2}\}$$

then we have that

$$A \sqcup B \coloneqq \{\text{ways to perform task 1 OR task 2}\}$$

Thus we see that we can apply the rule of sums whenever we are faced with *mutually exclusive options*. As an application of the rule of sum we can prove the generalized pigeonhole principle.

Proposition 1.4.8 (Generalized pigeonhole principle)**.** *If $m > kn$ then for every function $f : \underline{m} \to \underline{n}$, there is an element $h \in \underline{n}$ such that $|f^{-1}(h)| \geq k + 1$.*

Proof. Suppose that $|f^{-1}(h)| \leq k$ for every $h \in \underline{n}$. Since f is a function we can write

$$\underline{m} = \sqcup_{h=1}^{n} f^{-1}(h).$$

By the formal rule of sum we get

$$m = \sum_{h=1}^{n} |f^{-1}(h)| \leq kn.$$

■

Remark 1.4.9. Observe that this uses the concept of size, so implicitly assumes the Pigeonhole principle.

Formal rule of product

Proposition 1.4.10. *Let A and B be two finite sets, then*

$$|A \times B| = |A| \cdot |B|$$

Proof. Without loss of generality we can assume that $A = \underline{n}$ and $B = \underline{m}$ for some m and n. We reason by induction on n. If $n = 1$ then the assignment $(1, x) \mapsto x$ yields a bijection $\underline{n} \to \underline{m}$, and the statement is proven. Suppose now that the statement is true for some $n = k \geq 1$, we want to prove it for $n = k + 1$. We observe that we can write $\underline{k+1} \times \underline{m}$ as the disjoint union of two sets:

$$C_1 \coloneqq \{(a, b) \mid a \leq k\}$$

$$C_2 \coloneqq \{(k + 1, b) \mid a \leq k\}$$

By the rule of sums we have that

$$|A \times B| = |C_1| + |C_2|.$$

Now we observe that C_1 has the same size as $\underline{k} \times \underline{m}$ which we know to be (by the inductive hypothesis) km. At the same time we have that C_2 has the same size as $\underline{m}$. Thus we have that

$$|A \times B| = km + m = (k + 1)m.$$

■

The rule of product as stated in the book says

> If a task can be performed in n ways and a second task can be performed in m ways, then, provided that the two are performed simultaneously, we can performs the two tasks in nm ways

One can translate this in the formal rule of product by setting

$$A \coloneqq \{\text{ways to perform task 1}\}$$

$$B \coloneqq \{\text{ways to perform task 2}\}$$

then we have that

$$A \times B \coloneqq \{\text{ways to perform task 1 AND task 2}\}$$

Thus we see that we can apply the rule of product whenever we are faced with *simultaneous choices*.

1.5 Review of Counting Vocabulary

In this final section we review some basic vocabulary and facts about counting.

A **permutation** of size r of n objects is an injective map $\underline{r} \to \underline{n}$. The number of permutations of size r of n objects is denoted by $P(n, r)$ and we have that

$$P(n, r) = \begin{cases} \frac{n!}{(n-r)!} & \text{if } r \leq n \\ 0 & \text{otherwise.} \end{cases}$$

A **combination** of size r of n objects is a subset of $\underline{n}$ of size r. The number of permutations of size r of n objects is denoted by $C(n,r)$ and we have that

$$C(n,r) = \frac{P(n,r)}{r!} = \begin{cases} \frac{n!}{r!(n-r)!} & \text{if } r \leq n \\ 0 & \text{otherwise} \end{cases} =: \binom{n}{r}.$$

Theorem 1.5.1. *(Binomial Theorem)*

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}.$$

Proposition 1.5.2. *There are $\binom{n+r-1}{r}$ ways to chose r objects out of n types allowing repetitions.*

Given a non-negative integer n and non-negative integers $r_1, \dots r_k$ such that $r_1 + \cdots + r_k = n$ we define **multinomial coefficient**

$$\binom{n}{r_1 \dots r_k} := \frac{n!}{r_1! \cdots r_k!}.$$

] It computes the number of ways we can arrange n objects of k different types such that there are exactly r_i objects of type i.

LECTURE 2

Inclusion Exclusion

Lecture Plan

2.1 The principle of Inclusion Exclusion

Theorem 2.1.1 (Inclusion/Exclusion)**.** *Let S be a finite set and let A_1,...A_n be subsets of S. Then*

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{j=1}^{n} (-1)^{j+1} \alpha_j,$$

where

$$\alpha_j \coloneqq \sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} \left|\bigcap_{i \in I} A_i\right|$$

Before proceeding with the proof we have to introduce some tools. Given an universe set S and a set A the **characteristic function** (or indicator function) of A is the function $\mathbf{1}_A : S \to \mathbb{R}$ defined by

$$s \mapsto \begin{cases} 1 & \text{if } s \in A; \\ 0 & \text{if } s \notin A. \end{cases}$$

It is straightforward to see that

$$|A| = \sum_{s \in S} \mathbf{1}_A(s). \tag{2.1}$$

Proof of Theorem 2.1.1. Let us consider the function $f : S \to \mathbb{R}$ defined by

$$s \mapsto \sum_{j=1}^{n} (-1)^{j+1} \sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} \mathbf{1}_{\bigcap_{i \in I} A_i}(s).$$

Then, using (2.1), we have that

$$\begin{aligned}\sum_{s\in S} f(s) &= \sum_{s\in S}\sum_{j=1}^{n}(-1)^{j+1}\sum_{\substack{I\subseteq \underline{n}\\ |I|=j}} \mathbf{1}_{\bigcap_{i\in I} A_i}(s)\\ &= \sum_{j=1}^{n}(-1)^{j+1}\sum_{\substack{I\subseteq \underline{n}\\ |I|=j}}\sum_{s\in S} \mathbf{1}_{\bigcap_{i\in I} A_i}(s)\\ &= \sum_{j=1}^{n}(-1)^{j+1}\sum_{\substack{I\subseteq \underline{n}\\ |I|=j}}\left|\bigcap_{i\in I} A_i\right| = \sum_{j=1}^{n}(-1)^{j+1}\alpha_j\end{aligned}$$

Thus, to prove the statement we need only to show that $f = \mathbf{1}_{\bigcup_{i=1}^{n} A_i}$. To this aim, let $A := \bigcup_{i=1}^{n} A_i$ and $s \in S$, if s does not belong to A, then we have that s is not an element in any of the intersections $A_{i_1} \cap \cdots \cap A_{i_j}$ and thus $f(s) = 0$. Now we have just to show that $f(s) = 1$ when $s \in A$. Thus, let $s \in A$ and denote by $r = r(s) \leq n$ the number of the set A_i to which s belongs. Observe that s cannot belong to any intersection of the form $A_{i_1} \cap \cdots \cap A_{i_j}$ with $j > r$. On the other side, when $j \leq r$, s belongs to exactly $\binom{r}{k}$ intersections of the form $A_{i_1} \cap \cdots \cap A_{i_j}$. Using this, we can write

$$\begin{aligned}f(s) &= \sum_{j=1}^{n}(-1)^{j+1}\sum_{\substack{I\subseteq \underline{n}\\ |I|=j}} \mathbf{1}_{\bigcap_{i\in I} A_i}(s)\\ &= \sum_{j=1}^{r}(-1)^{j+1}\sum_{\substack{I\subseteq \underline{n}\\ |I|=j}} \mathbf{1}_{\bigcap_{i\in I} A_i}(s)\\ &= \sum_{j=1}^{r}(-1)^{j+1}\binom{r}{j}\\ &= (-1)\left(\sum_{j=0}^{r}(-1)^{j}\binom{r}{j} - (-1)^0\binom{r}{0}\right)\\ &(-1)\left((1+(-1))^r - 1\right) = 1, \qquad (2.2)\end{aligned}$$

where (2.2) is a consequence of the Binomial Theorem (Theorem 1.5.1).
Thus the proof is concluded. ■

If we set $\alpha_0 = |S|$, then we have an immediate corollary:

Corollary 2.1.2. *In the same assumption and notation of Theorem 2.1.1, we have*

$$\left|\bigcap_{i=1}^{n} A_i^c\right| = \sum_{j=0}^{n}(-1)^j\alpha_j.$$

Connection with notation used in the textbook

Let c be a condition on a set S, for example "$s \geq 2$", by $\overline{c}$ we denote the negation of c (if we keep the above example $\overline{c}$ is "$s < 2$". For conditions $c_1, \ldots c_n$, we introduce the

following notation

$$N(c_1, \dots, c_n) := |\{s \in S \mid s \text{ satisfies all the conditions } c_1, \dots c_n\}|.$$

By definition we also set $N(\emptyset) = |S|$ The principle of inclusion/exclusion as stated in the textbook is

$$N(\overline{c}_1, \dots, \overline{c}_n) = \sum_{j=0}^{n} (-1)^j \sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} N(c_i | i \in I).$$

Observe that this is a simple rewriting of Corollary 2.1.2. In fact, if we set

$$A_i := \{s \in S \mid s \text{ satisfies } c_i\},$$

we will have that $N(\overline{c}_1, \dots, \overline{c}_n)$ is precisely $|\bigcap_{i=1}^{n} A_i^c|$ and $N(\emptyset) = \alpha_0$, and, for $j > 0$,

$$\sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} N(c_i | i \in I) = \sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} \left| \bigcap_{i \in I} A_i \right| = \alpha_j.$$

2.2 Generalized principle of inclusion exclusion

In the notation above, we want to determine the sizes of the following subsets of S:

$$E_k := \{s \in S \mid s \text{ satisfies exactly } k \text{ of the conditions } c_i\}.$$

We have the following

Theorem 2.2.1. *In the notation above we have that*

$$|E_k| = \sum_{j=k}^{n} (-1)^{j-k} \binom{j}{j-k} \alpha_j,$$

where, as before,

$$\alpha_j = \sum_{\substack{I \subseteq \underline{n} \\ |I|=j}} N(c_i | i \in I)$$

Proof. This is Theorem 8.1 in the textbook. We refer to that. ∎

Observe that, if $k = 0$, we find again Corollary 2.1.2. Let now

$$L_k := \{s \in S \mid s \text{ satisfies at least } k \text{ of the conditions } c_i\}.$$

Then we have the following

Corollary 2.2.2. *In the previous notation*

$$|L_m| = \sum_{j=k}^{n} (-1)^{j-k} \binom{j}{k-1} \alpha_j$$

2.3 Applications

Applications to Arithmetic

The Euler ϕ-function is the function $\phi : \mathbb{N}_{\geq 2} \to \mathbb{N}$ defined by

$$\phi(n) \coloneqq |\{m \in \mathbb{N} \mid 1 \leq m \leq n, \gcd(m,n) = 1\}.|$$

Theorem 2.3.1. *The Euler ϕ-function can be expressed with the following law:*

$$\phi(n) = n \cdot \prod_{\substack{p|n \\ p \text{ prime}}} \left(1 - \frac{1}{p}\right)$$

Proof. We want to apply the principle of inclusion-exclusion on the universe set $S = \underline{n}$. To set up our problem, we let $p_1, \ldots, p_t$ be the prime divisors of n. Consider the following subsets of $\underline{n}$:

$$A_i \coloneqq \{s \in \underline{n} \mid p_i | s\}.$$

It is clear that

$$\left|\bigcap_{i \in I} A_i\right| = \frac{n}{\prod_{i \in I} p_i}.$$

Hence

$$\begin{aligned}
\phi(n) &= \left|\bigcap_{i=1}^{t} A_i^c\right| \\
&= \sum_{j=0}^{t} (-1)^j \sum_{\substack{I \subseteq \underline{t} \\ |I|=j}} \frac{n}{\prod_{i \in I} p_i} \\
&= n \cdot \frac{\sum_{j=0}^{t} (-1)^j \sum_{\substack{I \subseteq \underline{t} \\ |I|=t-j}} \prod_{i \in I} p_i}{\prod_{i=1}^{t} p_i} \\
&= n \cdot \frac{\prod_{i=1}^{t} (p_i - 1)}{\prod_{i=1}^{t} p_i} = n \prod_{i=1}^{t} \left(1 - \frac{1}{p_i}\right)
\end{aligned}$$

■

Derangements

Definition 2.3.2 (Derangement)**.** A **derangement** of n is a permutation $\sigma : \underline{n} \to \underline{n}$ such that $\sigma(i) \neq i$ for every $i \in \underline{n}$.

Proposition 2.3.3. *If we denote by d_n the number of derangements of n, we have that $d_n \approx n!e^{-1}$ when $n >> 0$. More precisely we have that*

$$|d_n - n!e^{-1}| \leq \frac{1}{n+2}.$$

Proof. We apply the principle of inclusion exclusion (Theorem 2.1.1) and we find that

$$\begin{aligned} d_n &= \sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)! \\ &= n! \sum_{k=0}^{n} (-1)^k \frac{1}{k!}. \end{aligned}$$

Thus

$$\begin{aligned} |n!e^{-1} - d_n| &= \left| n! \sum_{k=n+1}^{n} (-1)^{\infty} \frac{1}{k!} \right| \\ &\leq \frac{1}{n+1} - \frac{1}{(n+1)(n+2)} = \frac{1}{n+2}. \end{aligned}$$

■

LECTURE 3

Rook Polynomials

Lecture Plan

3.1 Rook Polynomials

Let C be a chessboard - a grid with cells, some of which are shaded (which are forbidden places). The k-th **rook number** of C is the number of ways in which we can place k rook in C such that no two rooks lies in the same row or column. By definition we assume that $r_0(C) = 1$ for any chessboard C.

Definition 3.1.1 (Rook Polynomial)**.** The **rook polynomial** of C is

$$r(C, x) \coloneqq \sum_{k=0}^{\infty} r_k(C) x^k.$$

Remark 3.1.2. Observe that this is indeed a polynomial as $r_k(C) = 0$ whenever k is bigger than minimum between the number of columns and the number of rows of C.

Definition 3.1.3 (Disjoint Union of Chessboards)**.** We say that a chessboard C is composed by the disjoint boards C_1 and C_2 if the boards C_i cover C and they have no column or row in common.

Proposition 3.1.4. *If C is composed of disjoint boards C_1 and C_2, then*

$$r(C, x) = r(C_1, x) \cdot r(C_2, x)$$

Proof. We have to show that the right-hand side and the left-hand side above have the same coefficients. To this end, we observe that the degree k coefficient in the left-hand side is exactly $r_k(C)$, which counts the way to place k rooks in C. Given k rooks, in order to place them in C we have to place some of them, say i in C_1 and the remaining $k - i$ in C_2. By the rule of products there are $r_i(C_1)r_{k-i}(C_2)$ ways to do that has the two placement are independent. By the rule of sum, we have that

$$r_k(C) = \sum_{i=0}^{k} r_i(C_1) r_{k-i}(C_2).$$

We conclude by observing that the latter expression is exactly the coefficient of degree k of $r(C_1, x) \cdot r(C_2, x)$. ■

Proposition 3.1.5. *Let C be a chessboard and F an allowed cell in C. Let C_s be the chessboard arising form C by removing the column and the row of F. Let C_e be the chessboard arising from C by forbidding F. Then*

$$r(C, x) = r(C_e, x) + xr(C_s, x).$$

Proof. As before we want to show that the degree k coefficients of the two expression coincide. To place k rooks in C there are two mutually exclusive cases

1. either we place a rook in F,
2. or we do not place a rook in F.

If the first, then we have to place the remaining $k-1$ rooks in C_s. If the latter then our placement of k-rooks in C is an admissible placemen of k rooks in C_e. By the rule of sum we get

$$r_k(C) = r_k(C_e) + r_{k-1}(C_s).$$

We conclude by observing that the right-hand side above is exactly the coefficient of degree k of $r(C_e, x) + xr(C_s, x)$. ■

Arrangements with forbidden positions

Let A and B be two sets with $|A| \leq |B|$. Given a family $\{B_a\}_{a \in A}$ of subsets of B indexed by A, we can build a chessboard C in the following way: C has rows indexed by A and columns indexed by B and the only allowed fields are of the form (a, b) with $b \in B_a$.

Proposition 3.1.6. *The number of injective functions $f : A \to B$ such that $f(a) \notin B_a$ for every $a \in A$ is*

$$\sum_{k=0}^{n} (-1)^k r_k(C) P(|A| - k, |B| - k).$$

Proof. Let S be the set of all injective functions $f : A \to B$ and denote by c_a the condition "$f(a) \in B_a$". To construct a function that satisfies at least k of this conditions we can choose k elements in A and their respective images in B_a, and then choose an injective functions between the remaining elements in A and B. There are $r_k(C)$ ways to make the first choice, and $P(|A| - k, |B| - k)$ ways to make the second choice. By the rule of product, we have that the number of functions that satisfies at least k of these conditions is

$$r_k(C) \cdot P(|A| - k, |B| - k).$$

By the inclusion-exclusion principle we have that

$$N(\bar{c}_a : |a \in A) = \sum_{k=0}^{n} (-1)^k r_k(C) \cdot P(|A| - k, |B| - k)$$

is the number of functions in S that satisfy none of the condition c_a, so such that $f(a) \notin B_a$ for every $a \in A$. ■

Exercise 3.1.7. Use the rook polynomials to compute the numbers d_3 and d_4.

LECTURE 4

Generating functions

Lecture Plan

4.1 Review of Power Series

A **formal power series** with real coefficients is a symbol

$$\sum_{n=0}^{\infty} a_n x^n$$

with $a_n \in \mathbb{R}$. A formal power series $S(x)$ has **positive radius of convergence** $\rho > 0$ if for every $|x| < \rho$ the sequence of functions

$$f_k(x) = \sum_{n=0}^{k} a_n x^n$$

converges absolutely to some function $f : (-\rho, \rho) \to \mathbb{R}$. When a power series is absolutely convergent, we have nice formulas to compute sum, products limits, derivatives and primitive for the limit function. In particular we have the following

Theorem 4.1.1. *Let $\sum_{n=0}^{\infty} a_n x^n$ and $\sum_{n=0}^{\infty} b_n x^n$ two power series absolutely converging to functions $f(x)$ and $g(x)$ with convergence radii ρ_1 and ρ_2. Then, for every $|x| < \min(\rho_1, \rho_2)$, we have*

1. *the series $\sum_{n=0}^{\infty}(a_n + b_n)x^n$ converges absolutely to $f(x) + g(x)$;*
2. *if we set $c_n := \sum_{k=0}^{n} a_k b_{n-k}$, the series $\sum_{n=0}^{\infty} c_n x^n$ converges absolutely to the function $f(x) \cdot g(x)$,*
3. *the series $\sum_{n=1}^{\infty} n a_n x^{n-1}$ converges absolutely to $\frac{d}{dx} f(x)$.*

We conclude this review paragraph with an useful criterion for absolute convergence:

Proposition 4.1.2. *Suppose that there is a positive constant C such that $|a_n| < C^n$ for every $n >> 0$. Then we have that the power series $\sum_{n=0}^{\infty} a_n x^n$ converges absolutely for every $|x| < \frac{1}{C}$*

4.2 Generating functions

Definition 4.2.1 (Generating Series and Generating Functions)**.** Let $(a_n)_{n\in\mathbb{N}}$ be a sequence of real numbers. The **generating series** of the sequence is the formal power series

$$\sum_{n=0}^{\infty} a_n x^n.$$

If this, for some $|x| < \rho$, converges absolutely to a function $f(x)$, we say that $f(x)$ is the **generating function of the sequence** $(a_n)_{n\in\mathbb{N}}$.

Example 4.2.2. The rook polynomial of a chessboard C is the generating function of the sequence $(r_n(C))_{n\in\mathbb{N}}$.

Fix k a positive integer, the polynomial $(1+x)^k$ is the generating function of the sequence $\left(\binom{k}{n}\right)_{n\in\mathbb{N}}$.

If $a_n = 1$ for $n \leq k$ and $a_n = 0$ otherwise. Then we have that the generating functions of $(a_n)_{n\in\mathbb{N}}$ is

$$\sum_{n=0}^{k} x^n = \frac{1-x^{k+1}}{1-x}.$$

If $a_n = 1$ for every n, we have that the generating series of the sequence (a_n) is $\sum_{n=0}^{\infty} x^n$. This converge absolutely to the function $f(x) = \frac{1}{1-x}$ for every $|x| < 1$.

If $a_n = n$, the generating series of $(a_n)_{n\in\mathbb{N}}$ is

$$\begin{aligned}\sum_{n=0}^{\infty} nx^n &= 0x^0 + \sum_{n=1}^{\infty} nx^n \\ &= \sum_{n=1}^{\infty} nx^n \\ &= x\left(\sum_{n=1}^{\infty} nx^{n-1}\right).\end{aligned}$$

Since $\sum_{n=0}^{\infty} x^n$ converge absolutely to $\frac{1}{1-x}$, by Theorem 4.1.1, we have that $\sum_{n=1}^{\infty} nx^{n-1}$ converges absolutely to $\frac{d}{dx}\frac{1}{1-x} = \frac{1}{(1-x)^2}$. We conclude that the generating function of $(n)_{n\in\mathbb{N}}$ is

$$f(x) = \frac{x}{(1-x)^2}.$$

If $a_n = n^2$, the generating series of $(a_n)_{n\in\mathbb{N}}$ is

$$\begin{aligned}\sum_{n=0}^{\infty} n^2 x^n &= \sum_{n=1}^{\infty} n^2 x^n \\ &= x\left(\sum_{n=1}^{\infty} n(n)x^{n-1}\right).\end{aligned}$$

Since $\sum_{n=0}^{\infty} nx^n$ converge absolutely to $\frac{x}{(1-x)^2}$, we reason as before and conclude that the generating function of $(n)_{n\in\mathbb{N}}$ is

$$f(x) = x\frac{d}{dx}\frac{x}{(1-x)^2} = \frac{x(x+1)}{(1-x)^3}.$$

4.3 Partitions

Compositions

Definition 4.3.1 (Composition of a natural number)**.** Given n a natural number, a **composition of** n is a way to write it as a sum

$$n = n_1 + \cdots + n_k$$

such that the n_i's are positive integers and the *order matters* (that is 1+2 and 2+1 are two different compositions of 3).

We want to compute $c(n)$, the number of composition for n for every positive natural number. In order to do that, we will first compute, for given n and k positive integers, the number c_n^k of composition of n with exactly k summands. We observe that this is the precisely the coefficient of degree n of the polynomial

$$(x + x^2 + x^3 + \cdots x^n)^k.$$

As higher degree terms do not change the coefficient of degree n, this is exactly the coefficient of degree n of the (a priori formal) power series

$$\left(\sum_{n=0}^{\infty} x^{n+1}\right)^k.$$

Since $\sum_{n=0}^{\infty} x^{n+1}$ converges absolutely somewhere (we can, for example use Proposition 4.1.2), we have that this is, indeed, generating function of the sequence of compositions. Thus the generating function of the c_n^k for any fixed k is given by $\left(\frac{x}{1-x}\right)^k$. By the rule of sum (cf. Proposition 1.4.7) we have that

$$c(n) = \sum_{k=1}^{\infty} c_n^k,$$

that is $c(n)$ is the coefficient of degree n of the series

$$\sum_{k=1}^{\infty}\left(\frac{x}{1-x}\right)^k = \frac{x}{1-x}\sum_{k=0}^{\infty}\left(\frac{x}{1-x}\right). \tag{4.1}$$

Now observe that if $|x| < \frac{1}{2}$, we have that $\left|\frac{x}{1-x}\right| < 1$, and the series in (4.1) converges absolutely to

$$f(x) = \frac{x}{1-x}\frac{1}{1-\frac{x}{1-x}} = \frac{x}{1-2x}.$$

In particular we have that $f(x)$ the generating functions of the $c(n)$. On the other side we can write

$$f(x) = x\sum_{n=0}^{\infty}(2x)^n = \sum_{n=1}^{\infty} 2^{n-1}x^n,$$

thus we get

$$c(n) = 2^{n-1}$$

Partitions

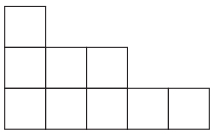
Definition 4.3.2 (Partition of a natural number)**.** Given n a natural number, a **partition of** n is a way to write it as a sum

$$n = n_1 + \cdots + n_k$$

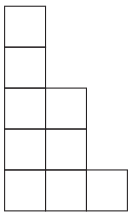
with n_i positive integers and the *order does not matter* (that is 1+2 and 2+1 give the same partition of 3). We denote by $p(n)$ the number of partitions of n.

Definition 4.3.3 (Young diagram)**.** Given $n = n_1 + \cdots n_k$ a partition of n, the Young diagram associated to it is a grid with k with row i consisting of n_i cells.

Example 4.3.4. The partition $9 = 1 + 3 + 5$ is represented by the diagram



Observe that, if we "transpose" - that is we swap rows and column and reorder appropriately - the above diagram we get



... which is the diagram corresponding to the partition $9 = 1 + 1 + 2 + 2 + 3$.

This example illustrates a general fact: by transposing the Young diagram of a partition of n with k summands we obtain the Young diagram of a partition of n in which every summand is at most equal to k (and at least one summand is equal to k). Thus we have the following

Proposition 4.3.5. *The number of partitions of n with at most k summands is equal to the number of partitions of n in which every summand is at most k.*

The remainder of this chapter is devoted to compute the generating function of the sequence $p(n)$. We have the following

Theorem 4.3.6. *The generating function of $(p(n))_{n\geq 1}$ is*

$$\prod_{k=1}^{\infty} \frac{1}{1-x^k}.$$

Proof. Given a partition of n, we denote by m_k the number of time the summand k appears. Then, we can write $n = \sum_{k=1}^{\infty} m_k \cdot k$. With this expression we can reason as

we did in the case of combinations, and see that the number of partitions of n with summands up to k is the coefficient of degree n of the product

$$\left(\sum_{j=0}^{\infty} x^j\right) \cdot \left(\sum_{j=0}^{\infty} x^{2j}\right) \cdots \left(\sum_{j=0}^{\infty} x^{kj}\right).$$

Thus the generating series of the $p(n)$ is obtained by taking the formal product of all the series involved, that is

$$\sum_{n=1}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \sum_{j=0}^{\infty} x^{kj}.$$

In order to find the generating function we have to show absolute convergence. Since the number of partitions is certainly less than the number of compositions, we have that $c(n) \leq 2^n$ and thus the left-hand side converges by Proposition 4.1.2. Thus we have to evaluate convergence on the right hand side. To this aim we take logarithms to transform a formal product in an infinite series. Without loss of generality we can suppose that $|x| < 1$ so that $\sum_{j=0}^{\infty} x^{kj}$ converges to $\frac{1}{1-x^k}$. Therefore, we get:

$$\begin{aligned}
\ln\left(\prod_{k=1}^{\infty}\sum_{j=0}^{\infty} x^{kj}\right) &= \ln\left(\prod_{k=1}^{\infty} \frac{1}{1-x^k}\right) \\
&= \sum_{k=0}^{\infty} \ln\left(\frac{1}{1-x^k}\right) \\
&= -\sum_{k=0}^{\infty} \ln\left(1-x^k\right).
\end{aligned}$$

Now if $|x| < c < 1$ for some c we have that

$$\left|\frac{d}{dx}\ln(1-x^k)\right| = \left|kx^{k-1}\frac{1}{1-x^k}\right| \leq kc^{k-1}\frac{1}{1-c}.$$

By Taylor formula we have that

$$|\ln(1-x^k)| \leq kc^{k-1}\frac{1}{1-c}|x|.$$

Thus we conclude that

$$\sum_{k=1}^{\infty} |\ln(1-x^k)| \leq \sum_{k=1}^{\infty} kc^{k-1}\frac{1}{1-c}|x| < +\infty.$$

Thus also the right-hand side converges and we can identify the generating functions, completing the proof of the statement. ■

LECTURE 5

Generating functions II

Lecture Plan

5.1 Exponential generating function

Definition 5.1.1 (Exponential generating function)**.** Let $(a_n)_{n\in\mathbb{N}}$ be a sequence of real numbers. The **exponential generating series** of the sequence is the formal power series

$$\sum_{n=0}^{\infty} \frac{a_n}{n!} x^n.$$

If this, for some $|x| < \rho$, converges absolutely to a function $f(x)$, we say that $f(x)$ is the **exponential generating function of the sequence** $(a_n)_{n\in\mathbb{N}}$.

Example 5.1.2. Fix k a positive integer. The polynomial $(1+x)^k$ is the exponential generating function of the sequence $(P(k,n))_{n\in\mathbb{N}}$.

If $a_n = 1$ for every n, we have that the generating series of the sequence (a_n) is $\sum_{n=0}^{\infty} \frac{1}{n!} x^n$. This converge absolutely to the function $f(x) = e^x$ for every x.

5.2 The summation operator

The function $\frac{1}{1-x}$ is called the **summation operation** in the context of generating functions. The reason behind the name is in the following result

Proposition 5.2.1. *Let $(a_n)_{n\in\mathbb{N}}$ a sequence with generating function $f(x)$. Then the function $\frac{f(x)}{1-x}$ is the generating function of the sequence $(\sum_{k=0}^{n} a_k)_{n\in\mathbb{N}}$*

Proof. Since the generating series of the sequence a_n converges, we can compute the generating function as a formal product of series. Then we have that, for sufficiently small x

$$\frac{f(x)}{1-x} = \left(\sum_{n=0}^{\infty} a_n x^n\right) \cdot \left(\sum_{n=0}^{\infty} x^n\right).$$

By Theorem 4.1.1, we have that the coefficient of degree n in the right-hand-side is precisely $\sum_{k=0}^{n} a_n$, and we conclude. ■

LECTURE 6

Recursion

Lecture Plan

6.1 First order recursion

Definition 6.1.1 (First order recurrence relation)**.** A **first order, linear, homogeneous recurrence relation with constant coefficients** is given by an expression of the form

$$a_{n+1} = da_n,$$

for every $n \in \mathbb{N}$, with $A \in$ a constant.

Fixing the initial value $a_0 = A$ is called **initial condition**. More generally, fixing the value of any a_{n_0}, for example setting $a_{n_0} = A$ is called **boundary condition**.

Proposition 6.1.2. *The unique solution of the first order problem $a_{n+1} = da_n$ with initial condition $a_0 = A$ is $a_n = Ad^n$*

Proof. We first show that the sequence $a_n = Ad^n$ satisfies the recurrence relation and the initial condition. For $n = 0$ we have that $a_0 = Ad^0 = A$, as the initial condition ask for. In addition for $n > 0$ we have that

$$a_n = da_{n-1} = d^2 a_{n-2} = \cdots = d^n a_0 = Ad^n,$$

as requested by the recurrence relation.

Now we want to show unicity. Suppose that there is another solution b_n to the recurrence problem. Suppose furthermore that both A and d are non zero, so that $a_n \neq 0$. Then we have that

$$\frac{b_n}{a_n} = \frac{db_{n-1}}{a_{n-1}} = \frac{b_{n-1}}{a_{n-1}} = \cdots = \frac{b_0}{a_0} = 1.$$

We conclude that $a_n = b_n$. If $d = 0$ we have that $a_n = 0$. On the other side since b_n satisfies the relation we have that $b_n = 0b_{n-1} = 0$. Thus, again $a_n = b_n$. Finally suppose that $A = 0$. Again we have $a_n = 0$ for every n. On the other side it is easy to prove by induction that $b_n = 0$ for every n. Again we have $b_n = a_n$ and the proof is completed. ■

6.2 Higher order recursion - the homogeneous problem

Definition 6.2.1 (Order k linear recurrence relation with constant coefficients)**.** A **order k, linear recurrence relation with constant coefficients** is given by an expression of the form

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = f(n),$$

with C_i real number and $f : \mathbb{R} \to \mathbb{R}$ a function. If $f \equiv 0$, we say that the recurrence relation is **homogeneous**.

Proposition 6.2.2. *Given a linear recurrence relation with constant coefficients*

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = f(n)$$

and a (particular) solution $(a_n^{(p)})$, then every other solution can be written as

$$a_n = a_n^{(p)} + a_n^{(h)},$$

where $a_n^{(h)}$ is any solution of the homogenous problem

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = 0$$

Proof. Directly using linearity. ■

Thus in order to find solution to a recurrence relation with constant coefficient we have to find a particular solution of the problem and the general solution of the associated homogeneous problem. In this Chapter we focus our attention to the latter. It turns out that there is a recipe to construct the general solution of an homogeneous problem by looking at the roots of a polynomial equation associated to the recursion relation.

Definition 6.2.3 (Characteristic equation of a linear recursion relation)**.** The **characteristic equation** of the homogeneous recurrence relation with constant coefficient.

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = 0$$

is the polynomial equation

$$C_k r^k + C_{k-1} r^{k-1} + \cdots + C_0 = 0$$

If a sequence of the form $a_n = Ad^n$ is solution of an homogeneous recurrence relation, then d is a root of the associated characteristic equation. The strategy behind the construction of the general solution of an homogeneous recurrence relation is to use the roots of the characteristic equation to produce linearly independent solutions. Any solution can be then be expressed as a linear combination of these. The main technical point is how to get a solution with real coefficients from complex roots, and how to produce the right number of independent solution from multiple roots. The following statement, whose proof we omit, solve the problem in the case of order 2 recurrence relations. The statement generalizes with minor modification to higher order relations.

Theorem 6.2.4. *Let $C_2 a_{n+2} + C_1 a_{n+1} + C_0 = 0$ a second order, homogeneous, linear, recurrence relation with constant coefficients. Then we have the following cases*

(a) *If the characteristic equation of the relation has* ***two distinct real roots****,* r_1 *and* r_2 *then every solution of the recurrence relation can be written as*

$$a_n = A_1 r_1^n + A_2 r_2^n$$

for A_1 *and* A_2 *two real numbers.*

(b) *If the characteristic equation of the relation has* ***two distinct complex roots,*** z_1 ***and*** $z_2 = \overline{z_1}$ *then every solution of the recurrence relation can be written as*

$$a_n = \rho^n \left(A_1 \cos n\theta + A_2 \sin n\theta\right)$$

for A_1 *and* A_2 *two real numbers,* $\rho = |z_1| = |z_2|$ *and* $\theta = \arg(z_1) \in [0, \pi]$.

(c) *If the characteristic equation has* ***one double real root****,* r*, then then every solution of the recurrence relation can be written as*

$$a_n = r^n (A_1 + A_2 n)$$

for A_1 *and* A_2 *two real numbers.*

LECTURE 7

Recursion II - The non homogeneous problem

Lecture Plan

7.1 The method of varying coefficients

Here there is not much theory to explain. The method of varying coefficients aims to provide a particular solution of an inhomogeneous recurrence relation like

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = f(n).$$

The main strategy is to perform an *educated guess* on the form of the solution, depending from the law of f. We allow some liberty in this educated guess (the varying coefficients). We plug in our guess in the problem and we find the right values of the coefficients. A list of educated guesses is provided in the textbook (Table 10.2 at page 479). The one tricky point is that one has to be careful that the educated guess is not a solution of the homogeneous problem. This happens, for example when $f(n) = ar^n$, with r a root of the characteristic polynomial. If this happens, one produce a solution to the non homogeneous problem by multiplying the educated guess by the minimum power of n such that the function obtained is not a solution of the homogeneous problem anymore. For example, if r is a root with multiplicity of of the characteristic equation, the right guess for the solution of the inhomogeneous problem with $f(n) = Ar^n$ is $a_n = Bn^2 r^n$ with B the varying coefficient.

7.2 The method of generating functions

In this section, we introduce another method to provide a particular solution of the following recurrence relation

$$C_k a_{n+k} + C_{k-1} a_{n+k-1} + \cdots + C_0 a_n = f(n). \tag{7.1}$$

with boundary conditions

$$a_0 = A_0,\ a_1 = A_1,\ \ldots, a_{k-1} = A_{k-1}.$$

If we multiply by x^{n+k} both sides of (7.1) and take the sum for all $n \geq 0$ we get

$$C_k \sum_{n=0}^{\infty} a_{n+k} x^{n+k} + C_{k-1} \sum_{n=0}^{\infty} a_{n+k-1} x^{n+k} + \cdots + C_0 \sum_{n=0}^{\infty} a_n x^{n+k} = \sum_{n=0}^{\infty} f(n) x^{n+k}.$$

which we can rewrite as follows

$$C_k \left(\sum_{n=0}^{\infty} a_n x^n - \sum_{n=0}^{k-1} A_n x^n \right) + C_{k-1} x \left(\sum_{n=0}^{\infty} a_n x^n - \sum_{n=0}^{k-2} A_n x^n \right) + \cdots + C_0 x^k \sum_{n=0}^{\infty} a_n x^n = \\ = x^k \sum_{n=0}^{\infty} f(n) x^n. \quad (7.2)$$

Assuming that all the series involved converges, we can denote by $G(x)$ the generating function of the sequence a_n and by $F(x)$ the generating function of the sequence $f(n)$. Then we can rewrite (7.2) in the following way:

$$C_k \left(G(x) - \sum_{n=0}^{k-1} A_n x^n \right) + C_{k-1} x \left(G(x) - \sum_{n=0}^{k-2} A_n x^n \right) + \cdots + C_0 x^k G(x) = x^k F(x). \quad (7.3)$$

By doing some algebra we get that

$$G(x) = \frac{x^k F(x) + \sum_{i=0}^{k} C_i \left(\sum_{n=0}^{i-1} A_n x^n \right)}{\sum_{i=0}^{k} C_i x^{k-i}}.$$

Now the main difficulty is to find the a_n from their generating function G. This can be done using the method of partial fractions (the same you use to compute the primitive of rational functions) when $F(x)$ is nice. In general one should compute the McLaurin expansion for G, but it is not always easy to find a law giving all of its coefficients. We refer to the book, which illustrates plenty of worked out examples.

Remark 7.2.1. Observe that the above discussion yields that the generating function of the solution of the homogeneous problem is

$$H(x) = \frac{\sum_{i=0}^{k} C_i \left(\sum_{n=0}^{i-1} A_n x^n \right)}{\sum_{i=0}^{k} C_i x^{k-i}}.$$

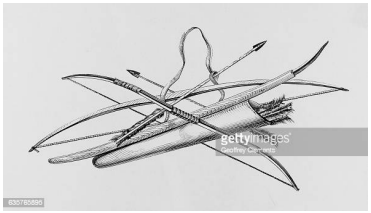
LECTURE 8

Graph Theory

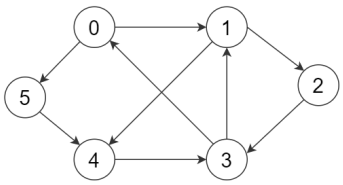
Lecture Plan

8.1 Definitions

Definition 8.1.1 (Directed graph)**.** A (simple)* **directed graph**, or **quiver**, G is given by a pair (V, E), where V is a finite set and $E \subset V \times V$.
The set V is the set of **vertices** of G, while the elements of E are said to be **edges**.



(a) Quiver



(b) Directed graph

Figure 8.1: Quivers

Given a directed graph $G = (V, E)$ we have two maps $s : E \to V$ defined by $(v_1, v_2) \mapsto v_1$ and $r : E \to V$ defined by $(v_1, v_2) \mapsto v_2$. We call s the **source** map (and v_1 the source of the edge (v_1, v_2)). On the other side r is the **range** map, and v_2 is the range of the edge (v_1, v_2).

Definition 8.1.2 (Undirected graph)**.** A (simple undirected) **graph**†, G is given by a pair (V, E), where V is a finite set and $E \subset \mathscr{P}(V)$ with $|E| = 1$ or 2.
Again the set V is the set of **vertices** of G, while the elements of E are said to be **edges**.

**simple* graph is opposed to multigraph - which will be defined later. If nothing is said all (directed or undirected graph will be considered simple)

†We follow the convention that the term graph, without additional adjectives, refers to a simple undirected graph.

Example 8.1.3. An important example of graph to keep in mind is K_n the complete graph on n vertices. We have that $V(K_n) = \underline{n}$ and

$$E(G) = \{\{i, j\} \mid i \neq j\}$$

Given a graph G, whether directed or undirected, we denote by $V(G)$ the set of vertices of G and $E(G)$ its set of edges. We say that two vertices v_1, v_2 are **adjacent** if there is an edges connecting them. That is

- for directed graphs we have that either (v_1, v_2) or (v_2, v_1) is in $E(G)$;
- for undirected graphs we have that $\{v_1, v_2\}$ is in $E(G)$.

An edge of the form (v, v) or $\{v\}$ is called a **loop**. A (directed) graph with no loops is called **loop free**.

Definition 8.1.4 (Walk)**.** Given a (simple undirected) graph G, a **walk** of length n from $x \in V(G)$ to $y \in V(G)$ is $(n+1)$-uple of vertices $(v_0, v_1, \ldots, v_n)$ such that

1. $v_0 = x$ and $v_n = y$;
2. for every $i \in \underline{n}$ we have that $\{v_{i-1}, v_i\}$ is in $E(G)$.

If $x = y$ we call the walk **closed**.

Exercise 8.1.5. Modify suitably the above definition to define a walk in a directed graph.

Definition 8.1.6 (Trail, circuit, path, cycle)**.** A walk on a graph G is said to be a

- **trail** if no edge is repeated;
- **circuit** if it is a closed trail;
- **path** if not vertex is repeated;
- **cycle** if it is a closed path.

Exercise 8.1.7. Modify suitably the above definition to define a trails, circuits, paths, and cycles in a directed graph.

Lemma 8.1.8. *Given a graph G, every walk from x to y with minimal length is a path.*

Proof. Let $(v_0, v_1, \ldots, v_n)$ a walk of minimal length from $x = v_0$ to $y = v_n$. Suppose that a vertex is repeated, that is $v_i = v_j$ for some $i, j \in \{0, n\}$. Up to relabelling we can suppose that $i \leq j$. Then the sequence $(v_0, \ldots, v_i, v_{j+1}, \ldots, v_n)$ yields a work from x to y of length $n - (j - i)$. The minimality assumption ensures that $i = j$ and thus the statement is proven. ■

Corollary 8.1.9. *If there is a walk in a graph between two vertices, then there is a path joining them.*

Definition 8.1.10 (Connected graph)**.** A graph G is said to be **connected** if there is a path between every two distinct vertices in $V(G)$.

Definition 8.1.11 (Multigraph)**.** A (undirected) **multigraph** is a triple (V, E, p) with V and E finite sets, and $p : E \to \mathscr{P}(V)$. a function such that $|p(E)| = 1, 2$.

Exercise 8.1.12. Modify this to give a definition of a directed multigraph.

8.2 Subgraphs, Complements and Isomorphism

Definition 8.2.1 (Subgraph)**.** Given a graph G a **subgraph** G_1 is a graph (V_1, E_1) such that the following hold:

1. $V_1 \subseteq V(G)$. In particular using the previous point we can identify $\mathscr{P}(V_1)$ with a subset of $\mathscr{P}(V)$.
2. With this identification in the previous point we have that $E_1 \subseteq E(G)$.

Definition 8.2.2 (Directed subgraph)**.** Given a directed graph G a (directed) **subgraph** G_1 is a directed graph (V_1, E_1) such that the following hold:

1. $V_1 \subseteq V(G)$. In particular using the previous point we can identify $V_1 \times V_1$ as a subset of $V \times V$.
2. With this identification in the previous point we have that $E_1 \subseteq E(G)$.

Exercise 8.2.3. Modify the above definition to find a definition of a subgraph of a multi graph.

We have an order relation on subgraphs of a given graphs: $G \subseteq G'$ if and only if G is a subgraph of G. Thus we can define:

Definition 8.2.4 (Connected component)**.** A **connected component** of a graph G is a maximal connected subgraph.

Definition 8.2.5 (Subgraph induced by U)**.** Given a graph $G = (V, E)$ and a subset $U \subseteq V$, we have that the **subgraph induced by** U is $(U, E \cap \mathscr{P}(U))$.

Exercise 8.2.6. Modify the above definition to find a definition of the directed graph induced by U when (V, E) is a directed graph.

Notation 8.2.7. If G is a (directed) graph and $v \in V(G)$ we denote by $G - v$ the (directed) graph induced by $V(G)\backslash\{v\}$. Given and edge $e \in E(G)$, $G - e$ will denote the (directed) graph $(V(G), E(G)\backslash\{e\})$.

Definition 8.2.8 (Complement of a graph)**.** Given a loopfree graph $G = (V, E)$ and a subgraph G', the **complement** of G' in G is $(v, \mathscr{P}(V)\backslash E)$.

Exercise 8.2.9. Modify the above definition to find a definition of complements for directed graphs.

Definition 8.2.10 (Graph homomorphism)**.** Given two graph G_1 and G_2 a **graphs homomorphism** $\varphi : G_1 \to G_2$ is given by a function $\varphi : V(G_1) \to V(G_2)$ such that $\varphi(e) \in E(G_2)$ for every $e \in E(G_1)$.

Exercise 8.2.11. Give the definition of morphisms for directed graphs.

Given a graph homomorphism $\varphi : G_1 \to G_2$ we say that it is an **isomorphism** if both the map induced on vertices and edges are bijective. We say that two graphs are **isomorphic** if there is an isomorphism between them.

Exercise 8.2.12. Show that graph isomorphism is an equivalence relation.

LECTURE 9

Graph Theory II

Lecture Plan

9.1 Euler circuits

Given a graph G and a vertex v, we say that an **edge** e is **adjacent** to v, and we write $e \sim v$ if $v \in E$ (if $e = (v, x)$ or $e = (x, v)$ for some $x \in v(G)$ in the case of directed graphs).

Definition 9.1.1 (Degree of a vertex)**.** Given a graph G and a vertex $v \in V(G)$ the **degree** of v, $\deg(v)$ is the number of edges adjacent to v (loop count twice)

A graph whose vertices have all the same degree d is called d**-regular**.

Proposition 9.1.2. *Let G be a graph of a multigraph, we have that*

$$2|E(G)| = \sum_{v \in V(G)} \deg(v)$$

Proof. The rough idea of the proof is that every vertices contribute 1 to the degree of each of his vertices (it works also for loops). Formally

$$\begin{aligned}
\sum_{v \in V(G)} \deg(v) &= \sum_{v \in V(G)} \sum_{\substack{e \in E(G) \\ e \sim v \\ \text{loops double}}} 1 \\
&= \sum_{e \in E(G)} \sum_{\substack{e \in E(G) \\ e \sim v \\ \text{loops double}}} 1 \\
&= \sum_{e \in E(G)} 2 = 2|E|.
\end{aligned}$$

■

Corollary 9.1.3. *In a graph or a multigraph, the number of vertices with odd degree is even.*

Definition 9.1.4 (Euler circuit)**.** Given a graph or a multigraph G, a circuit or trail in G is called **Euler** if it passes through all the edges.

Remark 9.1.5. We have not formally defined walk (and similar) notions on multigraph. Differently of what happens for a simple graph we have to keep track of edges. Thus a walk is a sequence $(v_0, e_1, v_1, e_2, v_2, \dots, e_n, v_n)$ such that $p(e_i) = \{v_{i-1}, v_i\}$.

Theorem 9.1.6. *Given a finite graph or multigraph G with no isolated vertices. We have that there exists an Euler circuit (trail) if, and only if, only one connected components of G has edges and every vertex in $V(G)$ has even degree (there are at most two vertices which have odd degree).*

Example 9.1.7. The finiteness hypothesis is necessary has the graph $(\mathbb{Z}, \{\{i, i+1\}, i \in \mathbb{Z}\})$ has all vertices of even degree but no Euler circuit.

The rest of this section will be devoted to prove Theorem 9.1.6 in the "circuit" case. We need a preliminary Lemma:

Lemma 9.1.8. *If G is a connected, finite, non-trivial, multigraph with all vertices of even degree, then there is a non-trivial circuit in G.*

Proof. By nontriviality there is at least one edge $e \in E(G)$. We will show that a trail of maximal length is an Euler circuit. To this aim let $(v_0, e_1, v_1, \dots, e_n, v_n)$ a trail of maximal length. If $v_n \neq v_0$, then we can extend it to a longer trail, since v_n has even degree. ■

Before proceeding with the proof of Theorem 9.1.6 we have to introduce the notion of **composition of walks** in a multigraph. Given two walks $W_1 = (v_0, e_1, v_1, \dots, e_n, v_n)$ and $W_1 = (v'_0, e'_1, v'_1, \dots, e'_l, v'_l)$ such that $v_n = v'_0$, we define the walk

$$W_1 \circ W_2 = (v_0, e_1, v_1, \dots, e_n, v_n, e'_1, v'_1, \dots, e'_l, v'_l).$$

Proof of Theorem 9.1.6. We first prove the sufficiency of the condition by induction on $|E|$. If $|E| = 0$, then, $V(G)$ is either empty or consisting of isolated points with no loops. If the first, it is impossible to negate the existence of an Euler circuit (or the empty circuit is an Euler circuit). If the latter, then (v) is an Euler circuit for every vertices in $V(G)$. Suppose now that the statement is true for $|E| \leq k$ for some non-negative integer k, we want to prove it for $|E| = k + 1$. By Lemma 9.1.8 there is a non-trivial circuit in G, $(v_0, e_1, v_1, \dots, e_n, v_n)$. By inductive hypothesis, every connected component of $G - e_1 - \cdots - e_n$ admits an Euler circuit E_i, "starting" at some v_i with $i \in \underline{t}$. Denote by $i_1, \dots, i_t$ the starting points of these circuit, then we have that

$$(v_0, \dots, v_{i_1}) \circ E_{i_1} \circ (v_{i_1+1}, \dots v_{i_2}) \circ E_{i_2} \circ \cdots \circ E_{i_t} \circ (v_{i_t}, \dots, v_n)$$

is an Euler circuit.

To prove necessity of the conditions, we may assume that G is loop free and that we have an Euler circuit for G $(v_0, e_1, v_1, \dots, e_n, v_n)$. As this visits all the edges of G, and there are not isolated vertices, then we have that it visits all the vertices of G. In particular G is connected. Let now $v \in V(G)$, then we have that

$$\deg(v) = |\{i \in \underline{n} \mid e_i \sim v\}|$$

$$= |\{i \in \underline{n} \,|\, v \in p(e_i) = \{v_{i-1,i}\}\}|$$
$$= 2\,|\{i \in \underline{n} \,|\, v = v_i\}|\,,$$

where in the first equality we used the loopfreeness. It follows directly that the degree of every vertices is even. ■

9.2 Planar graphs

In this section we will discuss whether a Graf can be drawn over a sheet of order without intersecting edges. This property is called planarity and it is geometric rather than combinatorial in nature. For this reason the material exposed here has some deep connection with the topology class, connections that are swiped under the carpet by our textbook. I will try to give hints in this direction. If you have taken Topology... please spend some time reflecting on this. If you have not taken topology, when you will please vote back to these pages and your eyes will be opened.

Given a graph $G = (V, E)$ we can choose an orientation of its edges and get a directed graph $\tilde{G} = (V, \tilde{E})$.

Definition 9.2.1 (Geometric realization of a graph)**.** The **geometric realization** of a graph $G = (V, E)$ is the metric space (X_G, d) where

$$X_G \coloneqq V \sqcup \left(\sqcup \tilde{E} \times [0,1]/\sim\right),$$

with $\tilde{E}$ an orientation of the edges in E and $\sim$ the equivalence relation generated by $(e, 0) \simeq s(e)$ and $(e, 1) = r(e)$. As distance we take the distance induced by the Euclidean metric on $[0, 1]$.

Remark 9.2.2 (Connection with Topology)**.** The space X_G is really a *1-dimensional cell complex*, where the vertices are the 0-dimensional cells and the edges are the one dimensional cells. We see that two edges intersect always in vertices (if they do intersect) - this gives exactly the definition of 1-dimensional cell complex.

We say that a graph G is **planar** if there is a continuous injective map $|G| \to \mathbb{R}^2$.

Definition 9.2.3 (Homeomorphic graphs)**.** Two graph are called **homeomorphic** if their geometric realizations are homeomorphic metric spaces. Equivalently, there exist a continuous bijective map between their geometric realizations such that the inverse is also continuous.

Theorem 9.2.4. *Two graphs are homeomorphic if, and only if, they admit isomorphic refinements by an elementary subdivision, that is replacing an edge $\{u, v\}$ with two edges $\{u, w\}$ and $\{w, v\}$ and V with $V \cup \{w\}$.*

Remark 9.2.5 (Connection with Topology)**.** The statement is clear if one think about cell complexes. The subdivision of the statement is just a subdivision of the cell complex. Topology says that two cell complexes have homeomorphic underlining spaces if and only if they are isomorphic (as cell complexes) after subdividing them.

Definition 9.2.6 (Bipartite graph)**.** A graph $G = (V, E)$ is bipartite if there is a partition $V = V_1 \sqcup V_2$ such that no edge is entirely contained in one of the two subset V_i.

Given n_1 and n_2 two positive integers we have that the **complete bipartite graph** K_{n_1,n_2} has vertices $\{(1,j),(2,k) \mid j \in \underline{n_1}; k \in \underline{n_2}\}$ and edges $\{(1,j),(2,k)\}$ for all j and k.

Theorem 9.2.7 (Kuratowski's Theorem)**.** *A graph G is not planar if, and only if, it contains a subgraph homeomorphic to either K_5 or $K_{3,3}$.*

We are not going to prove this theorem, but we will show that K_5 and $K_{3,3}$ are not planar. We begin with the following theorem.

Theorem 9.2.8. *Let G be a connected planar graph with v vertices and e edges. Consider an immersion $i : |G| \to \mathbb{R}^2$, and let f be the number of connected components of $\mathbb{R}^2 \backslash |G|$. Then we have that*

$$2 = v - e + f \tag{9.1}$$

Remark 9.2.9 (Connection with topology)**.** We are going in a moment to present a sketch of a completely combinatorial proof of this result. However, the deep reason why the results hold is topological. In fact, via the immersion i (extended to the Alexandroff one-point compactification of $\mathbb{R}^2$) we can see $|G|$ as the 1-dimensional skeleton of a cell complex subdivision of the Riemann sphere $\mathbb{S}^2$. What are the faces? Every cycle in the graph determine boards a face. Observe that a cycle also determines a connected component of the complement of $|G|$. There is a remaining *unbounded* connected component which give the last face of the subdivision. Thus the subdivision has exactly f faces. But know we use that we know that the *topological* Euler characteristic of the sphere is 2 and that, thanks to Algebraic topology, we can compute this with the formula on the right-hand side of (9.1).

Before proceeding with the proof we need the following vocabulary :

Definition 9.2.10 (Terminal vertex)**.** A vertex in a graph is said to be **terminal** if it has degree 1.

Sketch of the proof of Theorem 9.2.8. We reason by induction on $n = v + e$. If $n = 1$ then we have that G is either a point with no hedges, or a point with a loop (remember G is connected). In the first case $f = 1$, in the second $f = 2$. Thus in both we have $v + e - f = 2$. Suppose now that the result is true for $n \leq k$, we want to prove it for $n = k + 1$. We split our argument is some different cases.

Case 1: there is a loop l. If there is an edge l forming a loop we consider the graph $H := G - l$. This is a subgraph of G so is planar, it is connected and the number of connected components of its complement in $\mathbb{R}^2$ is $f - 1$. By induction hypothesis we have that

$$2 = v - (e - 1) + (f - 1) = v - e + f,$$

as we wanted.

Case 2: there is a terminal vertex x. We consider the graph $H := G - x$. This is a subgraph of G so is planar, it is connected and the number of connected components of its complement in $\mathbb{R}^2$ is f. By induction hypothesis we have that

$$2 = (v - 1) - (e - 1) + f = v - e + f,$$

as we wanted.

Case 3: no loops nor terminal vertices. Let l be any edge of G and consider the graph $H \coloneqq G - l$. If H is connected, then it satisfies the condition of the theorems. The number of connected components of the complement of H in $\mathbb{R}^2$ is $f - 1$. Then we have that

$$2 = v - (e - 1) + (f - 1) = v - e + f.$$

Otherwise suppose that H is not connected. Then it has 2 connected component (that in G are joined by l) H_1 and H_2 let v_i, e_i and f_i the number of vertices, edges and connected component of the complementary in $\mathbb{R}^2$ of the graph H_i. Then we have that $v_1 + v_2 = v$, $e_1 + e_2 = e - 1$ and $f_1 + f_2 = f + 1$ (the unbounded component is counted twice). By inductive hypothesis we have that $2 = v_i - e_1 + f_i$. Thus we have

$$4 = v_1 + v_2 - e_1 - e_2 + f_1 + f_2 = v - (e - 1) + f + 1 = v - e + f + 2.$$

Therefore $v - e + f = 2$ as we wanted. ■

Corollary 9.2.11. *In the notation above Let G a loop-free connected planar simple graph. Then $e \leq 3v - 6$. If in addition G is bipartite we have that $e \leq 2v - 4$.*

Proof. If G is a connected planar graph with no loops, then every connected component of its complement in $\mathbb{R}^2$ is bounded by at least 3 edges. Since every edges touches two components, we have that $2e \geq 3f$. Then we have that

$$2 = v - e + f \leq v - e + \frac{2}{3}e,$$

and so $e \leq 3v - 6$.

If, in addition G is planar, we have that any region is bordered by at least 4 edges, thus $e \leq 2f$

$$2 = v - e + f \leq v - e + \frac{1}{2}e.$$

We conclude that $e \leq 2v - 4$. ■

Example 9.2.12. Both K_5 and $K_{3,3}$ are not planar.

LECTURE 10

Graph Theory III

Lecture Plan

10.1 Hamiltonian cycles

Definition 10.1.1 (Hamiltonian cycle)**.** A path or cycle in a (multi)graph is said to be **Hamilton** if it visits every vertex.

Differently to what happens for Euler circuit, an intrinsic characterization of graph admitting a Hamiltonian cycle is not know. But we can give some sufficient condition

Theorem 10.1.2. *Let G be a loop free graph such that, for every two distinct vertices v and w we have that*

$$\deg(v) + \deg(w) \geq |V(G)| - 1.$$

Then G has a Hamiltonian cycle.

Proof. Omitted. ■

Example 10.1.3. The graph K_n has a Hamiltonian cycle if and only if $n \geq 3$.

Theorem 10.1.4. *Let G be a loop free graph with at least three vertices. If for every two non adjacent vertices v and w we have that*

$$\deg(v) + \deg(w) \geq |V(G)|,$$

then G has a Hamiltonian cycle.

Proof. We prove the contrapositive. Thus we assume that G has no Hamiltonian cycle and we show that there are two adjacent vertices such that

$$\deg(v) + \deg(w) < |V(G)|.$$

Denote by $n \geq 3$ the number of vertices of G. We can view G has a subgraph of K_n, the complete graph with n vertices. Since K_n has a Hamiltonian cycle, if we add edges to G we will eventually get to a graph which admits a Hamiltonian cycles. Let G' a subgraph of K_n containing G, maximal with respect to the property of not having a Hamiltonian cycle. Let $e = \{v, w\}$ an edge in the complement of G' then $G' + e$ admits a Hamiltonian cycle $(v_1, v_2, \ldots, v_n, v_1)$. This cycle must go through the edge e, thus, up to rotating, we can assume the that $v_1 = v$ and $v_2 = w$. For every $i = 3, \ldots, n$ we have that either $\{w, v_i\}$ or $\{v, v_{i-1}\}$ cannot be edges in G', otherwise the sequence

$$(w, v_i, v_{i+1}, \ldots, v_n, v, v_{i-1}, v_{i-2}, \ldots, w)$$

yields a Hamiltonian cycle in G'. In particular

$$\deg_{G'}(v) + \deg_{G'}(w) \leq \sum_{i=3}^{n} 1 + 1 < n$$

We conclude by observing that for every vertex v_i we have that $\deg_G(v_i) \leq \deg_{G'}(v_i)$.

■

Corollary 10.1.5. *If G is a loop-free graph such that*

$$|E(G)| \geq \binom{|V(G)| - 1}{2} + 2,$$

then G has a Hamiltonian cycle.

Proof. Again denote by $n \coloneqq |V(G)|$. Let v and w two non-adjacent vertices, then $G - v - w$ has $n - 2$ vertices and $|E(G)| - \deg(v) - \deg(w)$ edges. Since no loop-free graph on n vertices can have more edges than the corresponding complete graph, we have that

$$|E(G)| - \deg(v) - \deg(w) \leq \binom{n-2}{2}.$$

Suppose

$$|E(G)| \geq \binom{n-1}{2} + 2.$$

Then we have

$$\begin{aligned}
\deg(v) + \deg(w) &\geq |E(G)| - \binom{n-2}{2} \\
&\geq \binom{n-1}{2} + 2 - \binom{n-2}{2} \\
&= \frac{(n-1)(n-2)}{2} - \frac{(n-2)(n-3)}{2} + 2 \\
&= \frac{(n-2)(n-1-(n-3))}{2} + 2 \\
&= \frac{(n-2)2}{2} + 2 \\
&= n - 2 + 2 = n.
\end{aligned}$$

We conclude by applying Theorem 10.1.4. ■

The following is still open and widely contested

Conjecture (Lovàaz)**.** *Every vertex transitive* finite graph has a Hamiltonian cycle.*

10.2 Coloring

Chromatic Number

Definition 10.2.1 (Chromatic number)**.** A **proper** (or **admissible**) coloring of a (multi) graph G with n coloring is a function $f : V(G) \to \underline{n}$ such that $f(w) \neq f(v)$ whenever v and w are adjacent. The **chromatic number** of a (multi)graph G is

$$\chi_G \coloneqq \min\{n \in \mathbb{N} \mid \text{there exists a proper coloring of } G \text{ with } n \text{ colors}\}.$$

Proposition 10.2.2. *Given a graph G, it admits a proper coloring with n colors if, and only if, there is a graph morphism $f : G \to K_n$.*

Proof. Suppose that there is a graph morphism $f : G \to K_n$, the corresponding maps between vertices $f : V(G) \to \underline{n}$ sends adjacent vertices into adjacent vertices. Since K_n ha no loop, two adjacent vertices cannot have the same image, so f is a proper coloring.

Conversely, suppose that there is a proper coloring $f : V(G) \to \underline{n}$. Given to adjacent vertices v and w we have that $f(v)$ and $f(w)$ are distinct. In particular there is the edge $\{f(v, f(w)\}$ in K_n and we can extend f to a graph morphism. ■

Proposition 10.2.3. *The chromatic number of a graph is at most 2 if, and only if G is bipartite.*

Proof. If G is bipartite then we can color vertices belonging to different partitions in different colors. Thus the coloring number is at most 2.

Conversely, suppose that the coloring number is at most 2, then there is a graph morphism $f : G \to K_2$. Consider the corresponding function on vertices $f : V(G) \to \underline{2}$. By setting $V_1 \coloneqq f^{-1}(1)$ and $V_1 \coloneqq f^{-1}(2)$ we get a partition of $V(G)$. Two vertices are adjacent if and only if $f(v) \neq f(w)$. In particular they are adjacent if and only if they belong to different parts of the partition. Thus G is bipartite. ■

Chromatic Polynomial

Definition 10.2.4 (n-chromatic number)**.** Given a graph G, its **n-chromatic number** is

$$\chi_G(n) = |\{\text{proper colorings of } G \text{ with } n \text{ colors}\}|.$$

Theorem 10.2.5. *Given a graph G, there is a unique polynomial $P(G, x)$ such that, for every $n \in \mathbb{N}$, $n \geq 1$ we have that*

$$P(G, n) = \chi_G(n)$$

*A graph G is said to be vertex transitive if for every two vertices there is an automorphism of G exchanging them.

Before proceeding with the proof, we need to introduce the following construction. Given a graph G and $e \in E(G)$ we construct the graph G_e **obtained by collapsing the edge** e. We set

$$V(G_e) \coloneqq V(G)/_{v=w},$$

and

$$E(G_e) \coloneqq (E(G)\backslash\{e\})/_{v=w}.$$

Lemma 10.2.6. *Suppose that a graph G has connected component $G_1, \ldots, G_k$. Then*

$$\chi_G(n) = \chi_{G_1}(n) \cdots \chi_{G_k}(n)$$

Proof. Since vertices in different components are not adjacent, the colorings of the components are completely independent from one another and we can use the rule of product. ■

Proof of Theorem 10.2.5.

Proof of existence: By Lemma 10.2.6 we can assume that G is connected. If G has a loop, then no coloring is possible and we can take $P(G, x) \equiv 0$. Consequently, we can assume that G has no loop. We now reason by induction on $k = |V(G)| + |E(G)|$. If $k = 1$ then G consists of a vertex with no edges and we can take $P(G, x) = x$; we will have that $P(G, n) = n = \chi_G(n)$ for every $n \in \mathbb{N}$. We suppose now the result to be true for every $k \leq p$ and we prove it for $k = p + 1$. There are two cases:
Case a): G has a terminal vertex v. In this case we have that $\chi_G(n) = \chi_{G-v}(n)(n-1)$. By induction assumption we have that there is a polynomial $P(G - v, x)$ such that $\chi_{G-v}(n) = P(G - v, n)$, thus we can take

$$P(G, x) \coloneqq P(G - v, x)(x - 1). \tag{10.1}$$

Case b): G has no terminal vertex. Let $e = \{v, w\}$ be an edge in G (which is not terminal). For any proper coloring $f : V(G - e) \to \underline{n}$ there are two possibilities

1. either $f(v) = f(w)$, and thus f gives a proper coloring of G_e, or
2. $f(v) \neq f(w)$, and thus f yields a proper coloring of G.

By the rule of sums we have that

$$\chi_{G-e}(n) = \chi_G(n) + \chi_{G_e}(n).$$

By inductive hypothesis, there are two polynomials $P(G - e, x)$ and $P(G_e, x)$ realizing $\chi_{G-e}(n)$ and $\chi_{G_e}(n)$. Thus we can take

$$P(G, x) = P(G - e, x) - P(G_e, x). \tag{10.2}$$

Proof of unicity. Suppose that there are two polynomials $p(x)$ and $q(x)$ like in the statement. Then the polynomial $(p - q)(x)$ will have an infinite number of roots. Hence it will be the zero polynomial and we can deduce that $p = q$.

■

Definition 10.2.7 (Chromatic Polynomial)**.** The polynomial $P(G, x)$ appearing in the statement of Theorem 10.2.5 is called the **chromatic polynomial** of G.

Remark 10.2.8. Attention $P(G, x)$ is not the generating function for $\chi_G(n)$!!!

Remark 10.2.9. Equations (10.1) and (10.2) are crucial for computing the chromatic polynomial inductively.

Theorem 10.2.10. *Let G be a graph which is the union of two subgraphs G_1 and G_2 such that their intersection is K_n, then we have*

$$P(G, x) = \frac{P(G_1, x)P(G_2, x)}{P(K_n, x)}.$$

LECTURE 11

Trees

Lecture Plan

11.1 Basic definitions

Definition 11.1.1 (Tree)**.** A **tree** is a connected graph with no cycles. A graph with no cycles is called a **forest**.

Example 11.1.2. Let A_1,..., A_n be finite set with $A_i = \{a_{i1}, \ldots, a_{ik_i}\}$. We construct a tree T in the following way: $V(T) = \bigcup_{i \leq n} \times_{k=1}^{i} A_k$ - in which, by convention, we consider the empty product as a one point set $\{*\}S$; in addition we say that $(b_1, \ldots, b_k) \sim (b_1, \ldots, b_k, c)$ for every $c \in A_{k+1}$.

Exercise 11.1.3. Draw T when $n = 3$ and $A_i = \{0, 1\}$ for every $i = 1, 2, 3$.

Definition 11.1.4 (Spanning tree)**.** A **spanning tree** for a graph G is a subgraph T which is a tree and such that $V(T) = V(G)$.

Proposition 11.1.5. *Each connected graph has a spanning tree.*

Proof for finite graphs. . Let

$$\mathscr{T} \coloneqq \{T \subseteq G \textbf{ tree}\}.$$

This is an ordered set with respect of the inclusion of graphs. Since G is finite, it has finitely many subgraph so $\mathscr{T}$ is finite. Hence it has a maximal element $\overline{T}$. We need to show that $\overline{T}$ is a spanning tree. Suppose by contradiction that there is $v \in V(G)$ such that $v \notin V(\overline{T})$. Choose $w \in V(\overline{T})$. Since G is connected there is a path $(v_0, \ldots, v_n)$ with $v_0 = v$ and $v_n = w$. Let $n \geq k > 0$ minimal with the property that $v_k \in V(\overline{T})$. By adding the edge $\{v_{k-1}, v_k\}$ to $\overline{T}$ we get a bigger tree inside G, contradicting the maximality of $\overline{T}$. We conclude that that $v \in \overline{T}$. ■

Theorem 11.1.6. *The following are equivalent for a graph G:*

1. *G is a tree;*

2. *for every two distinct vertices v and w there is a unique path from v to w.*

If furthermore G is finite we have that these two conditions are equivalent to

3. *G is a connected and $|V(G)| = |E(G)| + 1$.*

Proof. Let $x \neq y$ twp vertices in G and assume that there are two different path from x to y, $(v_1, \dots, v_n)$ and $(w_1, \dots, w_k)$. Let

$$i \coloneqq \min\{j \,|v_j \neq w_j\},$$

and let

$$I \coloneqq \min\{j > i \, v_j = w_m \text{ for some } m > i\}$$

Then the walk $(v_{i-1}, v_j, \dots, v_I = w_k, w_{m-1}, \dots, w_{i-1} = v_{i-1})$ is a cycle in G, thus G is not a tree.
Conversely suppose that G is not a tree and let $(x, v_2, \dots, v_n, x)$ a cycle. Then (x, v_2) and $(x, v_n, v_{n-1}, \dots, v_2)$ are two different path between x and v_2.
Assume now that G is finite, we are going to show that 1. is equivalent to 3.
To this aim suppose that G is a tree, then it is either trivial or it has a terminal vertex. In fact, if $(v_1, \dots, v_n)$ is a path of maximal length in G we have that v_n can be adjacent only to v_{n-1}: it certainly it cannot be adjacent to one of the other v_is or the graph would have a cycle; it cannot be adjacent to a vertex not appearing in the path because this will contradict the maximality of the chosen path. We reason now by induction on the number of edges of G. If $|E(G)| = 0$, since G is connected, we have that G is trivial. Suppose that the statement is true for trees with at most k edges, and let us prove it for $|E(G)| = k + 1|$. Let e an extremal edge in G (that is the only edge connecting a terminal vertex to the rest of the graph). The graph $G - e$ is still a tree, and we have that $|E(G - e)| = k$ and $V(G - e) = V(G) - 1$. By induction we have that

$$|E(G)| = |E(G - e)| + 1 = |V(G - e)| - 1 + 1 = V(G) - 1$$

as we wanted.
Conversely, suppose now that $|V(G)| = |E(G)| + 1$. By Proposition 11.1.5, we know that G has a spanning tree T. We have that $E(T) \subseteq E(G)$ and

$$|E(T)| = |V(T)| - 1 = |V(G)| - 1 = |E(G)|.$$

Since $E(T)$ and $E(G)$ are finite sets, we conclude that $E(T) = E(G)$ and, in particular, G is a tree. ■

11.2 Spanning tree of a rooted tree

Definition 11.2.1 (Rooted tree)**.** A **rooted tree** is a pair (T, r) with T a tree and r a (distinguished) vertex of T, called **root**.

Example 11.2.2. In the tree of Example 11.1.2 it is natural to choose $*$ as root.

Definition 11.2.3 (Directed tree)**.** A **directed tree** is a directed graph whose underlining undirected graph is a tree.

Proposition 11.2.4. *Let (T, r) a rooted tree. There is only one orientation of its edge such that the incoming degree of v is 0 and the incoming degree of all the other vertices is 1.*

Proof for T finite. For any edge $\{v, w\}$ there are unique paths $(r, v_2, \ldots, v_n = v)$ and $(r, w_2, \ldots, w_k = w)$ connecting r with v and w respectively. Then either $v_{n-1} = w$ - in which case we orient $\{v, w\}$ as (w, v) - or $w_{k-1} = v$ - in which case we orient the edge as (v, w). This clearly gives an orientation of T in which the incoming degree of r is 0, or we would have a cycle in T. The unicity of the path connecting r with any vertices ensures that the incoming degree of any other vertex is 1. Suppose now that there is now that there is another orientation of the edges with the desired property, and let $(r = v_1, v_2, \cdots, v_n)$ a path such that $\{v_{n-1}, v_n\}$ is orientated differently in the two orientations. We can take it of be of minimal length with this property. Since in both orientations the incoming degree of r is 0 we have that $n \geq 3$. But then we will have that the incoming degree of v_{n-1} is at least 2 in the new orientation as both the edge $\{v_{n-2}, v_{n-1}\}$ and $\{v_{n-1}, v_n\}$ would be oriented in such a way that v_{n-1} is the range. Thus we get a contradiction. ■

Definition 11.2.5 (Leaf, child, parent)**.** In a tree a terminal vertex is called **leaf**. All other vertices are called **internal vertices**.
If (T, r) is a rooted tree and $v \in V(T)$ is any vertex, the **level of** v , $\mathrm{lv}(v)$ is the length of the unique path connecting r with v.
Given two adjacent vertices v and w, we say that v

- is a **child** of w if $\mathrm{lv}(v) > \mathrm{lv}(w)$;
- is a **parent** of w if $\mathrm{lv}(v) < \mathrm{lv}(w)$.

We can get the obvious notions of **descendants** and **ancestors** of a given vertex.

Lexicographical order

In order to run Depth-first type search algorithm on tree we have to assign an order to its vertices. A very natural order can be constructed as follows.

Consider first the rooted tree $(A, *)$ of Example 11.1.2 where $A_k = \{1, \ldots, m_k\}$. Then we say that

$$(a_1, \ldots, a_k) < (b_1, \ldots, b_l)$$

if one of the following conditions are satisfied

- either $k < l$ and $a_i = b_i$ for every $i < k$;
- or $I := \min\{i \,|a_i \neq b_i\} < k$ and $a_I < b_I$

A **lexicographical order** on a rooted tree (T, r) is an order induced by a graph homomorphism $f : T \to A$ which sends r to $*$.

Proposition 11.2.6. *Given any tree it admits a lexicographical order.*

Now given T a graph G and $\{v_1, \ldots, v_n\}$ and enumeration of its vertices, we can see in Algorithm 1 how one can run a depth first search in order to create a spanning tree.

Algorithm 1 Depth First algorithm to find a spanning tree

Require: $G = (V, E)$, $V = \{v_1, \ldots, v_n\}$
  $T \leftarrow (\{v_1\}, \emptyset)$
  $v \leftarrow v_1$
  if $\exists\, i$ such that $\{v, v_i\} \in E(G) \backslash E(T)$ **then**
    $T \leftarrow T + \{v, v_i\}$ ▷ Take i minimal with this property
    $v \leftarrow v_i$
  else
    Replace v by its parent.
  end if
  if $v = v_1$ **then**
  end if
  EXIT

LECTURE 12

Minimal spanning trees

Lecture Plan

12.1 Weighted graphs and shortest path

Definition 12.1.1 (Weighted graph)**.** A **weighted graph** is a graph $G = (V, E)$ together with a function $w : E \to \mathbb{R}^+$.
Given a path $p = (v_1, v_n)$ in a weighted graph (G, w), the **length of the path** is

$$l(p) \coloneqq \sum_{i=1}^{n-1} w(\{v_i, v_{i+1}\}).$$

Given a weighted graph (G, w) we can define a pseudo-distance

$$d : V(G) \times V(G) \to \mathbb{R}_{\geq 0} \cup \{\infty\}$$

in the following way

$$d(v, w) \coloneqq \inf\{l(p) \mid p \text{ is a path from } v \text{ to } w\},$$

where $l((v))$ is considered to be 0 and $\inf \emptyset$ is ∞

Proposition 12.1.2. *The function d defines a pseudo-distance on $V(G)$. That is the following are satisfied:*

1. $d(v, w) = 0$ *if and only if* $v = w$;
2. $d(v, w) = d(w, v)$ *for every* $v, w \in V(G)$; *and*
3. $d(v, w) \leq d(v, u) + d(u, w)$ *for every* $u, v, w \in V(G)$

Exercise 12.1.3. Prove the Proposition.

Remark 12.1.4 (Convention)**.** The weight function w can be extended to function, that we will still denote by w, $V(G) \times V(G) \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ by setting

$$w(v,w) = \begin{cases} w(\{v,w\}) & \text{if } \{v,w\} \in E(G) \\ \infty & \text{otherwise.} \end{cases}$$

We are now going to illustrate **Dijkstra shortest path algorithm** (see Algorithm 2) which has as *input* a weighted graph (G,w) and a starting vertex v_1. It gives as *output* a function

$$f : V(G) \to (\mathbb{R}_{\geq} \cap \infty) \times (V(G) \sqcup \{*\})$$

where $f(v) = (d(v_1,v), u(v))$ with $(v_1, \ldots, u^2(v), u(v), v)$ a shortest path from v_1 to v. If $f(v) = *$ we have that either $d(v_1,v) = 0$ and thus $v = v_1$ or $d(v,v_1) = \infty$ and there is no path from v_1 to v, that is v does not belong to the same connected component as v_1.

In explaining the algorithm we will denote $f(v) = (D(v), L(v))$

Algorithm 2 Dijkstra shortest path algorithm

Require: $G = (V,E)$, $v_1 \in V(G)$ and w weight function
  $S \leftarrow \{v_1\}$
  $D(v_1) \leftarrow 0$
  $D(v) = \infty$ for all other $v \neq v_1$
  $L(v) \leftarrow *$ for all v
  while $|S| < |V(G)|$ **do** ▷ Exit condition
    for $v \in V(G) \backslash S$ **do**
      if $\exists\, u \in S$ such that $D(u) + w(u,v) \leq D(v)$ **then**
        $D(v) \leftarrow \min_{u \in S}\{(D(w) + w(w,v), w)\}$
        $L(v) \leftarrow u$, with u such that $D(u) + w(u,v)$ is minimal.
      end if
    end for
    if For all $v \in V(G) \backslash S$ we have that $D(v) = \infty$ **then**
      EXIT ▷ This means that no replacement has taken place, thus we cannot reach any further vertex from v_1
    else
      $w \leftarrow v$ such that $v \in V(G) \backslash S$ and $L(v)$ is minimal
      $S \leftarrow S \cup \{w\}$
    end if
  end while

Remark 12.1.5. Observe that this algorithm give us a unique path connecting v_1 to any other vertex, thus if G is connected we get a spanning tree.

Idea of the proof of the correctness of the algorithm. The Algorithm terminates because either S grows at each iteration or the algorithm has a forced exit. The face that it provide with a shortest path can be proven by induction on the number of edges of the path returned. ■

Minimal Spanning Trees

Definition 12.1.6 (Minimal spanning tree)**.** Given (G, w) a weighted connected graph a **minimal spanning tree** is a spanning tree T for G such that

$$w(T) = \sum_{e \in E(T)} w(e)$$

is minimal.

We are going to give two algorithm to find a minimal spanning tree of a graph.

Kruskal's Algorithm

Kruskal's algorithm (see Algorithm 3) takes as *input* a weighted graph (G, w) and gives as output a subset $S \subseteq E(G)$ such that $(V(G), S)$ is a minimal spanning tree for G.

Algorithm 3 Kruskal's Algorithm

Require: $G = (V, E)$, and w weight function
  $S \leftarrow \emptyset$
  while $|S| < |V(G)| - 1$ **do** ▷ Exit condition
    Let $e \in E(G) \backslash S$, such that $w(e)$ is minimal and that $(V(G), S)$ is a forest.
    $S \leftarrow S \cup \{e\}$
  end while

Idea of the proof of correctness. As long as $|S| < |V(G)| - 1$ there is a vertex that is not "touched' by any edge in S, thus we can find an edge e such that $S \cup \{e\}$ is a forest. As S grows at every iteration the algorithm terminates.

It remain to show that the output is really a minimal spanning tree for G. To this aim, let $n = |V(G)|$ and $S = \{e_1, \dots, e_{n-1}\}$ where the edges have been enumerated following the order in which they had been "chosen' by the algorithm. Since $|S| = |V(G)| - 1$, by Theorem 11.1.6 Let T' a minimal spanning tree such that

$$d(T, T') \coloneqq \max\{i \mid \{e_1, \dots, e_i\} \subseteq E(T')\}$$

is maximal. We have to show that $d(T, T') = n - 1$, as this would imply that $T = T'$ and thus T is a minimal spanning tree.
Assume by contradiction that $d(T, T') = d < n - 1$, then the graph $T' + e_{d+1}$ contains a cycle since, again by Theorem 11.1.6, it cannot be a tree. Thus there is an edge e, lying on this cycle such that e is not an edge of T. Let $T'' = T + e_{d+1} - e$. We have that T'' is a spanning tree since it is connected and has n vertices and $n - 1$ edges. Now the edges $\{e_1, \dots, e_d, e\}$ in T' yield a forest, and by the choice of the algorithm we have necessarily that $w(e) \geq w(e_{d+1})$. Thus we get that

$$w(T'') = \sum_{f \in E(T'')} w(f) = \underbrace{\sum_{f \in E(T')} w(f)}_{=w(T')} \underbrace{-w(e_{d+1}) + w(e)}_{\leq 0} \leq w(T')$$

Since $W(T')$ is minimal, we get that $w(T'') = w(T')$ and necessarily $w(e_{d+1}) = w(e)$. But then $d(T, T'') = d + 1 > d(T, T')$ contradicting our choice of T'. ■

12.2 Prym's Algorithm

Prym's algorithm (see Algorithm 3) takes as *input* a weighted graph (G, w) and gives as output a subset $S \subseteq E(G)$ such that $(V(G), S)$ is a minimal spanning tree for G.

Algorithm 4 Prym's Algorithm

Require: $G = (V, E)$, $v \in V(G)$ and w weight function
  $S \leftarrow \emptyset$
  $P = \{v\}$
  while $|P| < |V(G)|$ **do** ▷ Exit condition
    Let $u \in P$, $w \in V(G) \backslash P$ such that $w(u, w)$ is minimal
    $P \leftarrow P \cup \{w\}$
    $S \leftarrow S \cup \{\{u, w\}\}$
  end while

LECTURE 13

Max flow and min cut theorem

Lecture Plan

13.1 Definitions and preliminary results

Definition 13.1.1 (Transport network)**.** A **transport network**, or simply network, is given by a pair (G, c), where G is a directed graph and $c : V(G) \times V(G) \to \mathbb{N}$ a function such that

1. there is a vertex a, called **the source**, with incoming degree 0;
2. there is a vertex z, called **the sink**, with outgoing degree 0;
3. the wight c, called **capacity** takes non-negative integer values and satisfies $c(v, w) = 0$ if $(v, w) \notin E(G)$.

Definition 13.1.2 (Flow)**.** Let $N = (G, c)$ a transport network with source a and sink z. A **flow** is a function $f : V(G) \times V(G) \to \mathbb{N}$ such that

1. $f(v, w) = 0$ if $(v, w) \notin E(G)$.
2. $f(v, w) \leq c(v, w)$ for every $(v, w) \in V(G)^2$;
3. (**Balancing condition**) for every vertex $v \neq a, z$ we have that

$$\sum_{w \in V(G)} f(v, w) = \sum_{w \in V(G)} f(w, v)$$

.

The last condition means that new material is created only at the source and material goes out just at the sink.

Definition 13.1.3 (Value of a flow)**.** Given a network $N = (G, c)$ with source a and sink z, and a flow f, the **value of the flow** f is

$$\mathrm{Val}(f) = \sum_{v \in V} f(a, w).$$

Our goal in this chapter is to give an algorithm to find a flow with a maximum value. To this aim we will need to introuduce the following notions:

Definition 13.1.4 (Cut)**.** A **cut** of a transport network $N = (G, c)$ is a two sets partition (P, P^c) of the vertices of G such that the source a is in P and the sink z is in P^c. Given a network N and a cut (P, P^c) the **capacity of the cut** is

$$\sum_{\substack{v\in P,\\ w\in P^c}} c(v,w).$$

We can use cuts to compute the value of a flow:

Lemma 13.1.5. *Let $N = (G, c)$ a network and (P, P^c) a cut. Given a flow f on N, we have that*

$$\mathrm{Val}(f) = \sum_{\substack{v\in P^c,\\ w\in P}} (f(w,v) - f(v,w))$$

Proof. Since the incoming degree of the source a is zero we can write

$$\mathrm{Val}(f) \coloneqq \sum_{v\in V(G)} f(a,v) = \sum_{v\in V(G)} (f(a,v) - f(v,a))$$

Now we huse the balancing condition and some algebra:

$$\begin{aligned}
\mathrm{Val}(f) &= \sum_{v\in V(G)} (f(a,v) - f(v,a)) \\
&= \sum_{v\in V(G)} (f(a,v) - f(v,a)) + \underbrace{\sum_{\substack{w\in P,\\ w\neq a}} \sum_{v\in V(G)} (f(w,v) - f(v,w))}_{=0} \\
&= \sum_{v\in V(G)} (f(a,v) - f(v,a)) + \sum_{v\in V(G)} \sum_{\substack{w\in P,\\ w\neq a}} (f(w,v) - f(v,w)) \\
&= \sum_{v\in V(G)} \left(f(a,v) - f(v,a) + \sum_{\substack{w\in P,\\ w\neq a}} (f(w,v) - f(v,w)) \right) \\
&= \sum_{v\in V(G)} \left(\sum_{w\in P} (f(w,v) - f(v,w)) \right) \\
&= \sum_{v\in P} \left(\sum_{w\in P} (f(w,v) - f(v,w)) \right) + \sum_{v\in P^c} \left(\sum_{w\in P} (f(w,v) - f(v,w)) \right) \\
&= \underbrace{\sum_{\substack{v\in P,\\ w\in P}} f(w,v) - \sum_{\substack{v\in P,\\ w\in P}} f(v,w)}_{\substack{=0,\\ \text{they cancel}}} + \sum_{\substack{v\in P^c,\\ w\in P}} f(w,v) - \sum_{\substack{v\in P^c,\\ w\in Pw\in P}} f(v,w) \\
&= \sum_{\substack{v\in P^c,\\ w\in P}} (f(w,v) - f(v,w))
\end{aligned}$$

■

As an easy corollary we see that we can compute the value of the flow using the sink, instead of the source:

Corollary 13.1.6. *Given a network $N = (G, c)$ with sink z, and a flow f we have that*

$$\mathrm{Val}(f) = \sum_{v \in V(G)} f(v, z).$$

Proof. Apply Lemma 13.1.5 to the cut $P = V(G)\backslash\{z\}$, $P^c = \{z\}$, and use that the outgoing degree of z is 0 and $f(z, z) = 0$ since the network has no loop. We get:

$$\begin{aligned}\mathrm{Val}(f) &= \sum_{v \neq z} \left(f(v, z) - \underbrace{f(z, v)}_{=0} \right) \\ &= \sum_{v \neq z} f(v, z) + \underbrace{f(z, z)}_{=0} \\ &= \sum_{vV(G)} f(v, z).\end{aligned}$$

■

13.2 The max flow min cut theorem

Another important application of Lemma 13.1.5 is that we can use it to compare the value of a flow with the capacity of a cut

Proposition 13.2.1. *Given a network $N = (G, c)$ with a cut (P, P^c) and a flow f, we have that*

$$\mathrm{Val}(f) \leq c(P, P^c).$$

Proof. We have that

$$\begin{aligned}\mathrm{Val(f)} &= \sum_{\substack{v \in P^c, \\ w \in P}} (f(w, v) - f(v, w)) \\ &\leq \sum_{\substack{v \in P^c, \\ w \in P}} f(w, v) \\ &\leq \sum_{\substack{v \in P^c, \\ w \in P}} c(w, v) = c(P, P^c).\end{aligned}$$

■

Observe that, if we find a flow f and a cut (P, P^c) such that

$$\mathrm{Val}(f) = c(P, P^c),$$

then we will have that $\mathrm{Val}(f)$ is maximal among all the values of possible flows, and at the same time $c(P, P^c)$ will be minimal among all the possible values taken by a capacity of a cut. The max flow and min cut theorem, states that this is alwasy the case: that is, a flow f has maximal value if and only if there is a cut such with capacity equal to $\mathrm{Val}(f)$. More precisely

Theorem 13.2.2 (Max Flow Min Cut Theorem)**.** *For a transport network N, the max value of a flow equals the min value of a capacity of a cut.*

The proof of this result will take the remainder of this lecture, and it will be divided in several substeps:

Lemma 13.2.3. *Let $N = (G, c)$ be a network with a flow f and a cut (P, P^c). Then* $\mathrm{Val}(f) = c(P, P^c)$ *if, and only if, the following conditions are satisfied:*

1. *for every $(v, w) \in E(G)$ such that $v \in P^c$ and $w \in P$ we have that $f(w, v) = c(w, v)$;*
2. *for every $(v, w) \in E(G)$ such that $v \in P^c$ and $w \in P$ we have that $f(v, c) = 0$.*

Proof. In the proof of Proposition 13.2.1 we had the following chain of inequalities:

$$\begin{aligned}\mathrm{Val(f)} &= \sum_{\substack{v\in P^c,\\ w\in P}} (f(w,v) - f(v,w)) \\ &\leq \sum_{\substack{v\in P^c,\\ w\in P}} f(w,v) \qquad (13.1)\\ &\leq \sum_{\substack{v\in P^c,\\ w\in P}} c(w,v) = c(P,P^c). \qquad (13.2)\end{aligned}$$

We have that $\mathrm{Val}(f) = c(P, P^c)$ if, and only if, both (13.1) and (13.2) are equalities. But (13.1) is an equality if and only if condition 1. holds, and (13.2) is an equality if, and only if, condition 2. holds. Thus we can conclude. ■

Definition 13.2.4 (Chain, backward and froward edges)**.** A **chain** on a directed graph G is path on the underlining undirected graph, that is it is a sequence of vertices $(v_1, \cdots, v_n)$ such that $v_i \sim v_{i+1}$ for every $i = 1, \ldots, n-1$. We say that an edge of a chain (v_i, v_{i+1}) is a **forward edge** if $(v_i, v_{i+1}) \in E(G)$. Otherwise, if $(v_{i+1}, v_i) \in E(G)$ we say that (v_i, v_{i+1}) is a **backward edge**.

Definition 13.2.5 (Augmenting path)**.** Given a network $N = (G, c)$ with source a and sink z, and a flow f, a chain $p = (v_1, \ldots, v_n)$ with $v_1 = a$ and $v_n = z$ is an **augmenting path for f** or an **f-augmenting path** if

1. $f(v_i, v_{i+1}) < c(v_i, v_{i+1})$ for every *forward edge*(v_i, v_{i+1}), and
2. $f(v_i, v_{i+1}) > 0$ for every *backward edge* (v_i, v_{i+1}).

Augmenting paths are crucial for our purpose. First if they do not exist, then we can produce a cut with the capacity of the flow value, thus we have a maximum flow.

Lemma 13.2.6. *Given a network $N = (G, c)$ and a flow f such that no f-augmenting path exist, then there is a cut (P, P^c) such that* $\mathrm{Val}(f) = c(P, P^c)$*. In particular* $\mathrm{Val}(f)$ *is maximal.*

Proof. Define the set P as follows

$$P \coloneqq \left\{ \begin{array}{l} v \in V(G) \mid \text{there is a chain } (v_1, \ldots, v_n) \text{such that:} \\ 1)\ v_1 = a,\ v_n = g; \\ 2)\ f(v_i, v_{i+1}) < c(v_i, v_{i+1}) \text{ for every forward edge } (v_i, v_{i+1}); \\ 3)\ f(v_i, v_{i+1}) > 0 \text{ for every backward edge } (v_i, v_{i+1}); \end{array} \right\}$$

Now $a \in P$ since we can consider the chain (a). On the other side $z \notin P$ since otherwise we would have an f-augmenting path in N. Thus we have a defined a cut. We shall now show that $c(P, P^c) = \mathrm{Val}(f)$ by applying Lemma 13.2.3. Take $v \in P$ and $w \in P^c$. Suppose first that $(v, w) \in E(G)$. Then we will have that $f(v, w) = c(v, w)$. Otherwise, consider a chain $(a, \ldots, v)$ in N such that

1. $f(v_i, v_{i+1}) < c(v_i, v_{i+1})$, for every forward edge (v_i, v_{i+1}), and
2. $f(v_i, v_{i+1}) > 0$, for every backward edge (v_i, v_{i+1}).

This exist because $v \in P$. If $f(v, w) < c(v, w)$ we can prolong the chain to the chain $(a, \ldots, v, w)$ which still satisfies 1. and 2. above, thus we would have that $w \in P$, reaching a contradiction.
In the same way, suppose now that $(w, v) \in E(G)$. If $f(w, v) > 0$, we can prolong a chain $(a, \ldots, v)$ to a chain $(a, \ldots, v, w)$ which still satisfies 1. and 2. above contradicting again that $w \in P^c$. In particular we have that the flow f and the cut P we just constructed, satisfy the condition of Lemma 13.2.3. Thus $\mathrm{Val}(f) = c(P, P^c)$ as we wanted. ■

This result tells us that, in order to find a maximum flow we have to modify a flow in such a way that no f-augmenting path exist. To this aim we introduce the following numerical invariants associated to a network a flow and an f-augmenting path.

Definition 13.2.7 (Δ_e)**.** Let $N = (G, c)$ a network, f a flow and $p = (v_1, \ldots, v_n)$ and f-augmenting path. For every edge $e = (v_i, v_{i+1})$ in p we set

$$\Delta_e \coloneqq \begin{cases} c(v_i, v_{i+1}) - f(v_i, v_{i+1}) & \text{if } e \text{ is forward} \\ f(v_i, v_{i+1}) & \text{if } e \text{ is backward.} \end{cases}$$

Observe that, since p is an f-augmenting path, we have that $\Delta_e > 0$. In particular if we set

$$\Delta_p \coloneqq \min\{\Delta_e \mid e \text{ is an edge in } p\},$$

we will have that $\Delta_p > 0$ for every p f-augmenting path. We will use this number Δ_p to increase the value of a given flow:

Lemma 13.2.8. *Consider a network $N = (G, c)$ with a flow f and an f-augmenting path p. Then the function $g : V(G) \times V(G) \to \mathbb{N}$ defined by*

$$g(v, w) \coloneqq \begin{cases} f(v, w) + \Delta_p & \text{if } (v, w) \text{ is a forward edge in } p \\ f(v, w) - \Delta_p & \text{if } (w, v) \text{ is a backward edge in } p \\ f(v, w) & \text{if } (v, w) \text{ is not an edge in } p \end{cases}$$

gives a flow on N with $\mathrm{Val}(g) = \mathrm{Val}(f) + \Delta_p$

Proof. We first check that g is a flow. Let (v, w) an edge in $E(G)$. If this does not appears in the chain we have that $g(v, w) = f(v, w) \leq c(v, w)$. If this appears in p as a forward edge, we have that

$$g(v, w) = f(v, w) + \Delta_p \leq f(v, w) + c(v, w) - f(v, w) = c(v, w).$$

Finally, if (v, w) is a backward edge in p then we have that $g(v, w) \leq f(v, w) \leq c(v, w)$. In any case we get that $g(v, w) \leq c(v, w)$. Next we have to check the balancing condition. Since the flow has been modified only in edges in $p$we have to see that this still holds for vertices which appears in p. Let, then, v_i be a vertex appearing in P $v_i \neq a$, z. There are 4 different cases, listed below and illustrated in Figure 13 at page 717 of the book.

1. Both (v_{i-1}, v_i) and (v_i, v_{i+1}) are forward edges in p. In this case we have that

$$\begin{aligned}\sum_{w \in V(G)} g(w, v_i) - g(v_i, w) = &\sum_{\substack{w \in V(G) \\ w \neq v_{i-1}, v_{i+1}}} (f(w, v_i) - f(v_i, w)) \\ &- (f(v_i, v_{i+1}) + \Delta_p) + \underbrace{f(v_{i+1}, v_i)}_{=0} \\ &- \underbrace{f(v_i, v_{i-1})}_{=0} + (f(v_{i-1}, v_i) + \Delta_p) \\ = &\sum_{w \in V(G)} f(w, v_i) - f(v_i, w) = 0.\end{aligned}$$

2. Both (v_{i-1}, v_i) and (v_i, v_{i+1}) are backward edges in p. In this case we have that

$$\begin{aligned}\sum_{w \in V(G)} g(w, v_i) - g(v_i, w) = &\sum_{\substack{w \in V(G) \\ w \neq v_{i-1}, v_{i+1}}} (f(w, v_i) - f(v_i, w)) \\ &+ (f(v_{i+1}, v_i) - \Delta_p) - \underbrace{f(v_i, v_{i+1})}_{=0} \\ &+ \underbrace{f(v_{i-1}, v_i)}_{=0} - (f(v_i, v_{i-1}) - \Delta_p) \\ = &\sum_{w \in V(G)} f(w, v_i) - f(v_i, w) = 0\end{aligned}$$

3. The edge (v_{i-1}, v_i) is forward and the edge (v_i, v_{i+1}) is backward. Then we have

$$\begin{aligned}\sum_{w \in V(G)} g(w, v_i) - g(v_i, w) = &\sum_{\substack{w \in V(G) \\ w \neq v_{i-1}, v_{i+1}}} (f(w, v_i) - f(v_i, w)) \\ &+ (f(v_{i+1}, v_i) - \Delta_p) - \underbrace{f(v_i, v_{i+1})}_{=0} \\ &- \underbrace{f(v_i, v_{i-1})}_{=0} + (f(v_i, v_{i-1}) + \Delta_p) \\ = &\sum_{w \in V(G)} f(w, v_i) - f(v_i, w) = 0\end{aligned}$$

4. The edge (v_{i-1}, v_i) is backward and the edge (v_i, v_{i+1}) is forward. Then we have

$$\begin{aligned}\sum_{w\in V(G)} g(w,v_i) - g(v_i,w) = &\sum_{\substack{w\in V(G)\\ w\neq v_{i-1},v_{i+1}}} (f(w,v_i) - f(v_i,w)) \\ &- (f(v_i,v_{i+1}) + \Delta_p) + \underbrace{f(v_{i+1},v_i)}_{=0} \\ &+ \underbrace{f(v_{i-1},v_i)}_{=0} - (f(v_i,v_{i-1}) - \Delta_p) \\ = &\sum_{w\in V(G)} f(w,v_i) - f(v_i,w) = 0\end{aligned}$$

Thus g also satisfies the balancing condition and thus it is a flow. We now have to compute the value of the flow. Observe that, since the ingoing degree of the source a is 0, we have that the edge (a, v_2) in the f-augmenting path p is forward. Therefore we get

$$\begin{aligned}\mathrm{Val}(g) &\coloneqq \sum_{v\in V(G)} g(a,v) \\ &= \sum_{\substack{v\in V(G)\\ v\neq v_2}} g(a,v) + g(a,v_2) \\ &= \sum_{\substack{v\in V(G)\\ v\neq v_2}} f(a,v) + f(a,v_2) + \Delta_p = \mathrm{Val}(f) + \Delta_p.\end{aligned}$$

■

We are now ready to conclude the proof of the max flow min cut theorem

Proof of Theorem 13.2.2. Fix (P, P^c) any cut. Since $\mathrm{Val}(f)$ is an integer less that $c(P, P^c)$, by Proposition 13.2.1 we have that the set

$$\{\mathrm{Val}(f) \mid f \text{ is a flow on } N\}$$

has a maximum value that we will denote by M. Let $\overline{f}$ a flow such that $M = \mathrm{Val}(\overline{f})$. By Lemma 13.2.8 we have that there is no augmenting path for $\overline{f}$, otherwise we could find a flow g with a higher value. We conclude, by applying Lemma 13.2.6, that there is a cut $(\overline{P}, \overline{P}^c$ such that

$$\mathrm{Val}(\overline{f}) = c(\overline{P}, \overline{P}^c.)$$

■

The strategy to use f-augmenting paths in order to modify the flow f to one with a bigger value can be implemented algorithmically. The Edmond-Karp Algorithm allows to find an f-augmenting path with the smallest possible number of edges. The Ford-Fulkerson Algorithm takes a flow as input, uses the Edmond-Karp Algorithm to find an f-augmenting path, compute the Δ_p of this path, and changes the flow accordingly. When no f-augmenting path can be found, it return the value of the flow and runs again the Edmond-Karp Algorithm to find the cut associated with the maximum flow.

LECTURE 14

Finite geometry and Latin squares

Lecture Plan

14.1 Finite Affine Planes

Definition 14.1.1 (Finite affine plane)**.** A **finite affine plane** is given by a pair $(\mathcal{P}, \mathcal{L})$, with $\mathcal{P}$ a finite set and $\mathcal{L} \subseteq \mathscr{P}(P)$ such that they verify the following axioms:

1. For every P and Q distinct elements in $\mathcal{P}$ there is a unique $l \in \mathcal{L}$ such that $l \supseteq \{p, q\}$.
2. (Euclid's V) For every $l \in \mathcal{L}$ and every $P \in \mathcal{P}$ with $P \notin l$ there is a unique l' in $\mathcal{L}$ such that $P \in l'$ and $l \cap l' = \emptyset$.
3. (Non degeneracy assumption) There are 4 distinct elements $P_1, \ldots, P_4$ in $\mathcal{P}$ such that there is no $l \in \mathcal{L}$ containing any 3 of them.

We call the elements of $\mathcal{P}$ **points**, while the elements of $\mathcal{L}$ are called **lines**.

Let $\mathbb{F}$ a finite field. Take a and b in $\mathbb{F}$ and consider the following subsets of $\mathbb{F}^2$:

- $l_{a,b} := \{(x, y) \in \mathbb{F}^2 \mid y = ax + b\}$, for a and b in $\mathbb{F}$, not both 0.
- $l_a := \{(a, y) \in \mathbb{F}^2\}$, for $a \in \mathbb{F}$

Theorem 14.1.2. *If $\mathbb{F}$ is not trivial, then the pair $\left(\mathbb{F}^2, \mathcal{L} := \{l_{a,b}, l_a \,|a, b \in \mathbb{F}\}\right)$, is an affine plane. We will denote it by $\mathbb{A}^2(\mathbb{F})$.*

Proof. First Axiom Let $P_i = (x_i, y_i)$, $i = 1, 2$, be two distinct points in $\mathbb{F}^2$. We consider separately the cases when $x_1 = x_2$ and $x_1 \neq x_2$.
If $x_1 = x_2$ clearly the line l_{x_1} contains both (x_1, y_1) and (x_2, y_2). We have to show that there is no other line that contains both of them. Clearly if $a \neq x_1$ then the line l_a does

not contain neither P_1 nor P_2. Suppose now by contradiction that there exist a line of the form $l_{a,b}$ containing both P_1 and P_2. Then we have

$$y_1 = ax_1 + b = ax_2 + b = y_2,$$

contradicting that P_1 and P_2 are taken to be distinct.
Suppose now that $x_1 \neq x_2$. Since $\mathbb{F}$ is a field and $x_1 - x_2 \neq 0_F$, there exist $(x_1 - x_2)^{-1} \in \mathbb{F}$. Let

$$a := (y_1 - y_2)(x_1 - x_2)^{-1}, \quad \text{and} \quad b := -x_1(x_1 - x_2)^{-1} + y_1.$$

Then one can easily verify that $l_{a,b}$ contains both P_1 and P_2. It is immediate to check that no line of the form $l_{a'}$ with $a' \in \mathbb{F}$ contains both P_1 and P_2, thus we only have to check that no other line of the form $l_{a',b'}$ contains both points. Suppose by contradiction that there is a line $l_{a',b'}$ containing both P_1 and P_2. The we would have

$$ax_1 + b = a'x_1 + b' \tag{14.1}$$
$$ax_2 + b = a'x_2 + b'. \tag{14.2}$$

By taking differences on both RHS and LHS of (14.1) and (14.2) we get that

$$a(x_1 - x_2) = a'(x_1 - x_2).$$

Since $\mathbb{F}$ is a field and by assumption $x_1 \neq x_2$ we get that $a = a'$. By plugging this inside (14.1) we have that

$$ax_1 + b = ax_2 + b'$$

and thus $b = b'$.

Euclid's V: Let l a line and $P = (x_1, y_1)$ a point outside l. We will construct the line l' passing through P and not intersecting l. We split our construction in two cases, considering separately when l is of the form l_a, and l is of the form $l_{a,b}$.
Suppose that $l = l_a$, then, since $P \notin l$ we have that $x_1 \neq a$. The line l_{x_1} passes through P and does not intersect l. Suppose that there is another line l' through P, that does not intersect l. Since l_{x_1} is the only line of this form that contains P, the l' has to be of the form $l_{a',b'}$. But this line contains the point $(a, a'a + b)$ which lies in l, giving us a contradiction.
Suppose now that l is of the form $l_{a,b}$. Since P is not in l, we have that

$$y_1 \neq ax_1 + b. \tag{14.3}$$

.The line $l_{a,y_1 - ax_1}$ passes through P. Suppose that this line intersect $l_{a,b}$ in a point (x_0, y_0), then we would have

$$b = y_0 - ax_0 = y_1 - ax_1,$$

contradicting (14.3). Thus we have constructed a line through P not intersecting l. We have to show that this is unique. Let l'' another line not intersecting l and containing P. As lines of the form $l_{a'}$ intersect l in the point $(a', aa' + b)$ we have necessarily that $l'' = l_{a',b'}$ for some a' and b' in $\mathbb{F}$. If $a \neq a'$, then the two lines intersect in the point $\big((b' - b)(a - a')^{-1}, a(b' - b)(a - a')^{-1} + b\big)$, thus $a = a'$. But then we have that $b' = y_1 - ax_1$, or otherwise l'' could not contain P, thus we have that $l'' = l'$, and the proof is complete.

Non degeneracy As a non-trivial field always contains two distinct elements 0_F and 1_F, we have that $\mathbb{F}^2$ contains at least four distinct elements: $(0_F, 0_F)$, $(1_F, 0_F)$, $(0_F, 1_F)$, and $(1_F, 1_F)$. We have that:

- $(0_F, 0_F)$ is contained in lines of the form l_{0_F} or $l_{a,0_F}$, with $a \neq 0$.
- $(0_F, 1_F)$ is contained in lines of the form l_{0_F} or $l_{a,1_F}$.
- $(1_F, 0_F)$ is contained in lines of the form l_{1_F} or $l_{a,-a}$ with $a \neq 0$.
- $(1_F, 1_F)$ is contained in lines of the form l_{1_F} or $l_{a,1_F-a}$. Thus we see that no line can contain 3 of these 4 points.

■

Example 14.1.3. In Figure 14.1 you can see depicted $\mathbb{A}(\mathbb{F}_3)$.

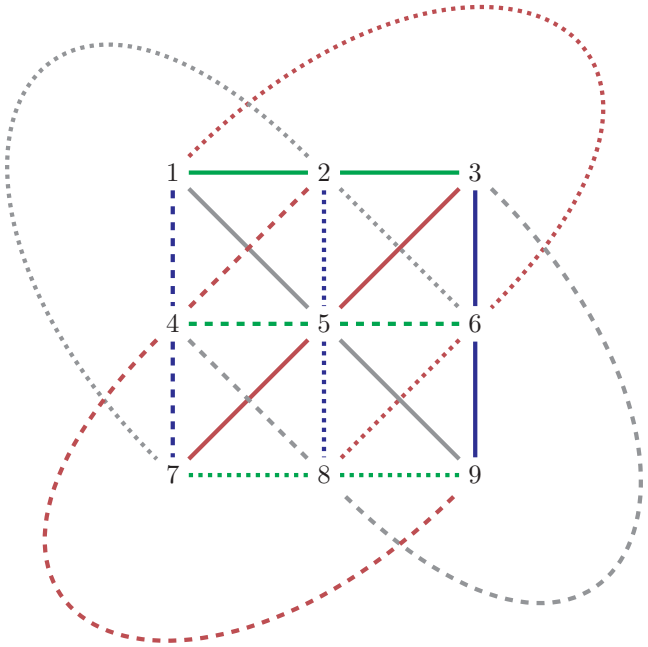


Figure 14.1: Affine plane on $\mathbb{F}_3$

Definition 14.1.4 (Parallel lines)**.** Given a finite affine plane $(\mathcal{P}, \mathcal{L})$ we say that two lines, l and l', are **parallel** if either $l = l'$ or $l \cap l' = \emptyset$. If l is parallel to l' we write

$$l // l'.$$

Proposition 14.1.5. *Being parallel is an equivalence relation.*

Proof. The reflexivity and symmetry of the relation are clear, thus we have just to prove transitivity. To this aim suppose that we have three line l_1, l_2 and l_3 such that l_1 is parallel to l_2 and l_2 is parallel to l_3. Suppose first that $l_1 = l_2$. Then clearly we have that l_1 is parallel to l_3. We can, then, assume that $l_1 \cap l_2 = \emptyset$. If $l_3 = l_2$ we can deduce immediately that $l_1 \cap l_3 = \emptyset$, and so that l_1 is parallel to l_3. Thus we can assume that $l_2 \cap l_3 = \emptyset$. Suppose by contradiction that there is a point $P \in l_1 \cap l_3$. As $P \notin l_2$, we have that both l_1 and l_3 are line through P that do not intersect l_2. By Euclid's V axiom we have that $l_1 = l_3$ and the proof is complete. ■

We call the equivalence classes of this equivalence relation **parallelity classes**. In Figure 14.1 lines in the same parallelity classes have been drawn in the same color (but with different styles). We notice that there are 4 parallelity classes, each of one containing 3 lines. Every line passes through 3 points and there is a total of 12 lines. This is a special case of the very general statement below.

Theorem 14.1.6. *Given a finite affine plane $X = (\mathcal{P}, \mathcal{L})$, then there is an integer $n \geq 2$ such that*

- *Every line in X contains exactly n points.*
- *Every point of X is contained in exactly n lines.*
- *There are exactly $n + 1$ parallelity classes in X, each containing exactly n lines.*
- *There are $n^2 + n$ lines in X.*
- *There are n^2 points in X.*

Definition 14.1.7 (Rank of a finite affine plane)**.** Given a finite affine plane X, the number n such that every line contains exactly n points is called the **rank** of X.

Before proceeding with the proofs, we need to show some preliminary results.

Lemma 14.1.8. *Every finite affine plane $X = (\mathcal{P}, \mathcal{L})$ has at least 3 parallelity classes.*

Proof. Let P_i, with $i = 0, \ldots 3$, be the four points prescribed by the non-degeneracy axiom. Let l_i denote the unique line through the points P_0 and P_i with $i = 1$, 2 and 3. We claim that the lines l_i are not parallel with each other. In fact, since they all contain the point P_0, if two of them were to be parallel, they would be equal. But this would imply that there would be a line in X through three of the point $P_0, \ldots, P_4$ contradicting the non-degeneracy axiom. ■

Lemma 14.1.9. *Let X be a finite affine plane. Suppose that a parallelity classes in X contains exactly m distinct lines. If l does not belong to the parallelity class then l contains exactly m points.*

Proof. Let $l_1, \ldots, l_m$ distinct line in the parallelity class. As l is not parallel to any of those we have that $l \cap l_i$ consist of exactly one point P_i, and the points P_i are distinct or the lines l_i cannot be distinct. By Euclid's V axiom we have that $P = \cup_{i+1}^{m} l_i$. In

fact let Q a point in X, if it is not in any of the l_i, then there should exist a new line l' containing Q and parallel to the l_i, and this would contradict the fact that the parallelity classes consists of m distinct lines. In particular we have that

$$l = l \cap X = \cup_{i=1}^{m} l \cap l_i = \{P_1, \dots, P_m\}.$$

■

Proof of Theorem 14.1.6. By Lemma 14.1.8 there are at least 3 parallelity classes in X. Let l_1, l_2, and l_3 be arbitrary lines in different parallelity classes. Denote by n_i the size of the line l_i, and m_i the size of the parallelity class of the line l_i.

By Lemma 14.1.9 we have that, for $i \neq j$, $m_i = n_j$. In particular $n_1 = m_2 = n_3$ and $n_1 = m_3 = n_2$. As the lines were taken to be arbitrary, we can conclude that $n_1 = n_2 = n_3$. We will denote this value simply by n.

Again as a consequence of Lemma 14.1.9, we get that $m_1 = m_2 = m_3 = n$. In particular every parallelity classes consists of n lines and in total there are n^2 points in X. Since $|X| \geq 4$ by the non-degeneracy axiom, we have that $n \geq 2$.

Now we construct $n + 1$ lines passing through a point $P \in X$. Let P a point in X and l a line through P. As $n \geq 2$ there is another line l', distinct from l, and parallel to it. For each $Q \in l'$ we consider l_Q, the only line containing both Q and P. These are pairwise distinct. In fact, suppose otherwise that $l_Q = l_{Q'}$ for distinct points Q and Q'. As l' is the only line through Q and Q' we would have that l' passes through P, contradicting our choice of l'. These are also distinct from the line l. If $l = l_Q$ for some point Q in l' we would have $l \cap l' \neq \emptyset$, contradicting again our choice of l'. As l' contains n points we have constructed $n + 1$ lines passing through P. We want to show that no other line contains P. But if l'' is a line through P, there are two cases: either it is parallel to l' or not. If the first, then, by Euclid's IV axiom, we have that $l = l''$. If the latter, then $l'' \cap l'$ will consists of just one point Q, and by the first axiom we will have that $l'' = l_Q$.

■

14.2 Latin squares

Definition 14.2.1 (Latin Square)**.** Let n a positive integer. A **Latin square of size** n is a table with n rows and n columns in which n different symbol appears. Every symbol must appears exactly once in every row and every column.

For our purposes, the set of available symbols will always be either $\{1, \dots, n\}$ or $\{0, 1, \dots, n-1\}$. We say that a Latin square is in **standard form** if its first row is $(1, 2, \dots, n)$ (or $(0, \dots, n-1)$ for the other choice of symbols)† Every Latin square can be reduced in standard form by permuting / relabelling the symbols.

Example 14.2.2. Suppose that the set of symbol is $S := \{0, 2, \dots, n-1\}$. Then the assignation

$$a_{ij} = i + j - 2 \mod n \quad \text{for } i, j = 1, \dots n$$

†Given any set of symbols S, and a complete ordering $<$, we say that a Latin square is in standard form if and only if the first row is $(s_1, \dots, s_n)$ with $s_1 < s_2 < \cdots < s_n$.

gives a Latin square in standard form. If furthermore m is an integer such that $\gcd(n, m) = 1$, then the assignation

$$b_{ij} = m(i-1) + j - 1 \mod n \quad \text{for } i, j = 1, \dots n$$

gives a Latin square in standard form.

Definition 14.2.3 (Orthogonal Latin Squares)**.** Two Latin squares $A = (a_{ij})$ and $B = (b_{ij})$ of size n are **orthogonal** if the set

$$\{(a_{ij}, b_{i,j} \mid i, j = 1, \dots, n\}$$

has size n^2.

This means that every pair of symbols (i, j) appears exactly once.

Theorem 14.2.4. *There are at most $n-1$ pairwise orthogonal Latin squares of size n in standard form.*

Proof. Omitted. ■

Remark 14.2.5. This is just an upper bound: for $n = 6$ there are no pairwise orthogonal Latin squares in standard form.

Producing orthogonal Latin squares with modular algebra

Theorem 14.2.6. *If $n = p^t$ with p a prime and $t > 1$ an integer, then there are exactly $n-1$ mutually orthogonal Latin squares in standard form.*

Proof. Let $\mathbb{F}$ be the finite field of size n, and let $\{f_1, \dots, f_n\}$ an enumeration of its elements such that $f_n = 1_{\mathbb{F}}$ and $f_n = 0_{\mathbb{F}}$. For $m = 2, \dots, n$ consider the matrix

$$L_m := (f_m \cdot f_i + f_j)_{i,j+1,\dots n}.$$

Since $\mathbb{F}$ is a field we have that L_m is a Latin square for every m. In fact the following cancellation laws apply:

- $f_m \cdot f_i + f_j = f_m \cdot f_i + f_k \quad \Rightarrow f_j = f_k$. Thus no element is repeated twice in a row;
- $f_m \cdot f_i + f_j = f_m \cdot f_k + f_j \quad \Rightarrow f_i = f_k$. Thus no element is repeated twice in a column.

The first row of each of these Latin square is $(f_j)_{j=1,\dots,n}$, so these are in standard form. Clearly the L_m are distinct: the element in position $(n, 0)$ is exactly f_m, and it varies for every m. In order to conclude our proof, we have only to check that these are mutually orthogonal. To see this we have to show that the following two equations holds simultaneously when $f_m \neq f_k$ if and only if $f_i = f_r$ and $f_j = f_s$:

$$\begin{cases} f_k \cdot f_i + f_j = f_k \cdot f_r + f_s & \text{(14.4a)} \\ f_m \cdot f_i + f_j = f_m \cdot f_r + f_s & \text{(14.4b)} \end{cases}$$

By subtracting (14.4a) to (14.4b) we get

$$(f_m - f_k)f_i = (f_m - f_k)f_r$$

Which yields $f_i = f_r$. By plugging this in into (14.4a) we get that $f_j = f_s$. ■

Remark 14.2.7. Observe that the proof teaches us how we can construct orthogonal Latin squares. Try to construct 4 orthogonal Latin squares of size 5.

Connection with finite geometry

Theorem 14.2.8. *Let $(\mathcal{P}, \mathcal{L})$ be a finite affine plane of rank n. If $c_0, \dots, c_1$ is an enumeration of its parallelity classes, we can identify every c_i with the set $\underline{n}$, and $\mathcal{P}$ with $\underline{n} \times \underline{n}$ by setting $P = (i, j)$ if and only if P is contained in the i-th line of c_0 and in the j-th line of c_1. For each $k = 1, \dots, n$ define the matrix $L^{(k)}$ by setting $(L^{(k)})_{ij} = l$ if, and only if, the point (i, j) is contained in the l-th line of c_k. Then the $L^{(k)}$ are $n - 1$ orthogonal Latin squares.*

Proof. Omitted. ■

Corollary 14.2.9. *There is no affine plane of rank 6.*