Tutorials: NDMI025 - Randomized Algorithms

Karel Král

April 15, 2021

This text is a work in progress, do not distribute. All errors in this text are on purpose. Please report them to my email `kralka@iuuk.mff.cuni`....

Contents

Chapter 1

Exercises

1.1 Tutorial

1. - Can you all hear me?
   - If you are uncomfortable asking a question in English, just ask in Czech/Slovak and I will translate.
   - Have you all taken:
     (a) a probability course (discrete probability, random variables, expected value, variance, Markov, Chernoff)
     (b) a linear algebra course (matrix operations, linear maps, eigenvectors and eivenvalues, discriminant)
     (c) a graph theory course (what a combinatorial graph is, bipartite, complete, coloring)
     (d) a combinatorics course (factorial, binomial coefficients)
     (e) an algorithms / programming course (big-O notation, possibly understanding Python based on the other question)
   - This class is heavy on theory. Are you interested in computer simulations and or implementations? If so:
     (a) Python
     (b) R
     (c) C++

   Solution: 1

2. You are presented with two sealed envelopes. There are k\$ in one of those and ℓ\$ in the other ($k, \ell \in \mathbb{N}$ but you do not know k, ℓ in advance). You may open an envelope and (based on what you see) decide to take this one or the other (without looking into both).
   (a) Is there a way how to walk away with the larger amoung of money with probability strictly larger than 0.5?
   (b) What is the expected value you walk away with (in terms of k, ℓ)?
   (c) Simulate.

   Solution: 2

3. Graph isomorphism. You have seen an interactive proof of graph non-isomorphism on the class. Can you come up with an interactive proof of graph isomorphism?

   Solution: 3

4. We will focus on random walks and their properties a lot.

   (a) Random walks are useful when analysing algorithms – "two coloring without monochromatic triangle" of three-colorable graph.

   (b) Random numbers in the computer are often expensive to generate, can we reduce number of used random bits (expanders)? Or even get a deterministic algorithm?

   (c) To sample from extremely large spaces.

   Let $n \in \mathbb{N}$, say $n = 30$. Let us the following problem we start with $X_0 = \lfloor n/2 \rfloor$ and do the following process:

   - if $X_i \in \{0, n\}$ we stop
   - we set $X_{i+1} = X_i + \delta$ where δ is picked uniformly at random from $\{-1, 1\}$

   (a) Is this a Markov chain (Definition 2.2)? If so can you write it's matrix?

   (b) What is the expected number of steps until stopping?

   Solution: 4

5. Think of some example MCs.

   (a) Create a MC that is irreducible.

   (b) Create a MC that is not irreducible.

   (c) Create a MC that is periodic.

   (d) Create a MC that is not periodic.

   (e) Compute a stationary distribution of the following MC:

   $$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

   (f) Create a MC that has more stationary distributions.

   Solution: 5

6. We are collectors and we want to collect all n kinds of coupons. Coupons are sold in packages which all look the same. Thus when we buy an coupon, we buy one of n kinds uniformly at random. This is known as the *coupon collector* problem.

   (a) What is the expected number of coupons we need to buy to get all kinds?

   (b) How many coupons do we need to buy to have probability at least $1 - q$ of collecting all kinds?

   (c) What is the Markov chain? Is this similar to a random walk on some graph?

   (d) Simulate.

   Solution: 6

1.2 Tutorial

1. Find a family of oriented graphs of constant in-degree and constant out-degree and as large hitting time as possible.

   Note that similar situation could happen on undirected graphs where the probabilities of traversing edge one way and the other way would not be the same. Which is in principle almost an oriented graph.

   Solution: 1

2. Let $A \in \mathbb{R}^{n \times n}$ be a matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$. Show that the matrix $A + dI_n$ has eigenvalues $d + \lambda_1, \ldots, d + \lambda_n$.

   Solution: 2

3. Show Courant-Fisher: Let $A \in \mathbb{R}^{n \times n}$ be a symmetric matrix ($A^T = A$). Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ be its eigenvalues. Show

   (a) $\lambda_1 = \max_{x \in \mathbb{R}^n, \|x\|=1} x^T A x$

   (b) $\lambda_n = \min_{x \in \mathbb{R}^n, \|x\|=1} x^T A x$

   (c) The eigenvalue λ_2 can be computed similarly $\lambda_2 = \max_{x \in \mathbb{R}^n, \|x\|=1, x^T u_1 = 0} x^T A x$ (where u_1 is the eigenvector corresponding to λ_1). We can get other eigenvalues in a similar manner. Moreover we could use this to prove the interlacing theorem. See `https://en.wikipedia.org/wiki/Min-max_theorem`

   Solution: 3

4. Show that a connected d-regular graph is bipartite iff the least eigenvalue of its adjacency matrix is $-d$.

   Solution: 4

5. Compute the eigenvalues and eigenvectors of the following graphs:

   (a) K_n, the complete graph on n vertices.

   (b) $K_{n,n}$, the complete bipartite graph with partites of size n each.

   (c) C_n, the cycle on n vertices.

   Solution: 5

1.3 Tutorial

1. You are given two coins. One is fair and the other one has $\Pr[\text{tails}] = 1/4$. We use the following algorithm to distinguish those:

   - Pick a coin and toss it n times.
   - Let $\hat{p}$ be the probability of getting a tails (number of tails over n).
   - If $\hat{p} \geq 3/8$ we say this coin is fair.

   Show that if $n \geq 32 \ln(2/\delta)$ then our algorithm answers correctly with probability at least $1 - \delta$.

   Solution: 1

2. You have seen that ZPP = RP $\cap$ co-RP.

   (a) Recall definitions of:

- RP
- ZPP
- co-RP
- BPP
- NP

(b) Show that RP $\subseteq$ NP (and thus co-RP $\subset$ co-NP).

(c) Decide if BPP = co-BPP.

(d) Show that if NP $\subseteq$ BPP then NP=RP.

Solution: 2

3. How to simulate a fair coin using a tipped coin and wice versa.

   (a) We are given a fair coin $\Pr[tails] = 0.5$. Show how to generate a random bit with $\Pr[1] = p$ for a given $p \in (0, 1)$ (both $p = 0$ and $p = 1$ are a bit boring).

   (b) We are given a tipped coin – we do not even know $p = \Pr[tails]$. We are sure that $\Pr[tails] \in (0, 1)$. Generate a fair coin toss.

   Solution: 3

4. Show that the expected number of comparisons a quick-sort algorithm does is roughly $n \ln(n)$. Show that probability of it making at least $32n \ln(n)$ comporisons is at most $1/n^3$.

   Solution: 4

1.4 Tutorial

1. We have k servers that are supposed to handle $n \gg k$ jobs. But the jobs come online and there is no single computer that knows the loads of servers (otherwise we would have a lot of communication). How do we distribute the jobs? We distribute the jobs each independently uniformly at random. How to bound the maximum load?

   Solution: 1

2. Distributed discrete logarithm algorithm (Breaking the Circuit Size Barrier for Secure Computation Under DDH, Boyle, Gilboa, Ishai linked on the website).

   Solution: 2

3. Let A, B be two disjoint sets of vertices where $|A| = |B| = n$. Let $d \geq 5$ be a constant. We choose d uniformly at random edges from each vertex from A. We show that with constant positive probability each set $S \subseteq A$ of size $|S| \leq n/d$ has at least $\beta|S|$ neighbors where $\beta = d/4$.

   Solution: 3

1.5 Tutorial

1. Let us define the *edge expansion* for a given graph G by:

$$h(G) = \min_{|S| \leq n/2} \frac{e(S, V \setminus S)}{|S|}$$

For any $S \subseteq V(G)$ we denote

$$e(S) = E(G) \cap S \times S = \text{number of edges inside } S$$
$$e(S, V(G) \setminus S) = E(G) \cap (S \times (V(G) \setminus S)) = \text{number of edges going from } S \text{ to the complement}$$

Let us show that if λ_2 is the second largest eigenvalue of the adjacency matrix of a d-regular graph G then:

$$h(G) \geq \frac{d - \lambda_2}{2}$$

Solution: 1

2. Show that for any $v \in \mathbb{R}^n$ it holds that

$$\frac{1}{\sqrt{n}} \|v\|_1 \leq \|v\|_2 \leq \|v\|_1$$

Solution: 2

3. Let μ be a probability distribution, that is $\|\mu\|_1 = 1$ and $\mu_j \geq 0$ (for each $j \in \Omega$). Let us define $d(\mu, \nu)$ the distance of two probability distributions as:

$$d(\mu, \nu) = \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)|$$

Show that:

$$d(\mu, \nu) = \max_{A \subseteq \Omega} \mu(A) - \nu(A)$$

where $\mu(A) = \sum_{x \in A} \mu(x)$.

Solution: 3

4. Let M be a Markov chain on the set of states S. We say that a Markov chain $Z_t = (X_t, Y_t)$ on the set of states $S \times S$ is a coupling iff

$$\Pr[X_{t+1} = x' \mid Z_t = (x, y)] = \Pr[M_{t+1} = x' \mid M_t = x]$$
$$\text{(where } X_{t+1} \text{ is the first coordinate of } Z_{t+1})$$
$$\Pr[Y_{t+1} = y' \mid Z_t = (x, y)] = \Pr[M_{t+1} = y' \mid M_t = y]$$
$$\text{(where } Y_{t+1} \text{ is the second coordinate of } Z_{t+1})$$

So one can imagine a coupling as a Markov chain, that in both coordinates behaves in the same way as the original Markov chain (but the coordinates might be dependent on each other).

Let $Z_t = (X_t, Y_t)$ be a coupling of a Markov chain M on the state space S. Suppose there is a T such that:

$$\Pr[X_T \neq Y_T \mid X_0 = x, Y_0 = y] \leq \varepsilon \qquad \text{(for all } x, y \in S)$$

then

$$\tau(\varepsilon) \leq T$$

Where formally the mixing time $\tau(\varepsilon)$ is defined as

$$p_x^t = \text{ the distribution when starting at } x \text{ and doing } t \text{ steps}$$

$$\tau(\varepsilon) = \max_{x \in S} \min \left\{ t \mid d(p_x^t, \pi) \leq \varepsilon \right\}$$

Notice that when we prove that $\Pr[X_T \neq Y_T \mid X_0 = x, Y_0 = y] \leq \varepsilon$ for all $x, y \in S$, we know that we are close to the stationary distribution (without even knowing the stationary distribution).

Solution: 4

5. We define the hypercube graph of dimension d as follows: the vertices are binary strings of length d and two vertices are connected by an edge iff they differ in exactly one coordinate. For instance $d = 2$ the graph is

$$(\{00, 01, 10, 11\}, \{(00, 01), (00, 10), (11, 01), (11, 10)\})$$

(the edges are not oriented).

We start at O^d and do the following random walk:

- With probability $1/2$ we stay at the current vertex.
- With probability $1/2$ we choose uniformly at random and index $j \in [d]$ and change the corresponding bit.

The Markov chain is nice (it converges to a single stacionary distribution, namely the uniform distribution on all vertices). Our question is how many steps do we need to take until we are "close enough" to the uniform distribution. Show that the random walk has $\tau(\varepsilon) \leq d \ln(d/\varepsilon)$.

Solution: 5

1.6 Tutorial

1. Definition: a random variable – our estimate $A > 0$ is an $\varepsilon - \delta$ approximation of a value $g > 0$ if

$$\Pr\left[(1 - \varepsilon)g \leq A \leq (1 + \varepsilon)g\right] \geq 1 - \delta$$

Prove the *Estimator Theorem*: Let U be a finite set and $G \subseteq U$ its subset. We know $|U|$ and wish to estimate $|G|$. If we take n uniformly random and independent samples from U where

$$n \geq \frac{3}{\varepsilon^2 \frac{|G|}{|U|}} \ln(2/\delta)$$
$$X = \text{ number of samples inside of } G$$

and output $A = X\frac{|U|}{n}$ then A is $\varepsilon - \delta$ approximation of $|G|$.

Solution: 1

2. We say that $\hat{x}$ is an ε-aproximation of x iff

$$(1 - \varepsilon)x \leq \hat{x} \leq (1 + \varepsilon)x$$

Show that for $\varepsilon < 1/2$ if we have ε-aproximation $\hat{s}$ of a number s and ε-aproximation $\hat{t}$ of a number t then $\hat{s}/\hat{t}$ is an 4ε-aproximation of s/t.

Solution: 2

3. Let $\varepsilon > 0$ be fixed. Find a suitable choice of $\bar{\bar{\varepsilon}}$ such that if we take $(\hat{a_i})_{i=1}^n$ of numbers $(a_i)_{i=1}^n$ then $\prod_{i=1}^n \hat{a_i}$ is an ε-aproximation of $\prod_{i=1}^n a_i$.

   Solution: 3

4. Show an algorithm that given a bipartite graph G (partites consisting of the same number of vertices) determines if the number of perfect matchings is even or odd.

   Solution: 4

5. Let $A \in \{0,1\}^{n\times n}$ be a matrix. Let $\varepsilon_{i,j}$ be independent random ± 1 variables. Let $B \in \{-1,0,1\}^{n\times n}$ be a matrix such that $B_{i,j} = \varepsilon_{i,j} A_{i,j}$ (uniformly randomly independently assign signs to entries of A).

   (a) Show that $\mathbb{E}[\det(B)] = 0$

   (b) Show that $\mathbb{E}[\det(B)^2] = \text{perm}(A)$ (permanent of A)

   Solution: 5

6. Let $G = (U \cup V, E)$ be a bipartite graph such that $|U| = |V| = n$ and $\delta(G) > n/2$ (the least degree). Show that for any matching of size at most $n-1$ there is an augmenting path of length at most 3.

   Solution: 6

7. Let $G = (U \cup V, E)$ be a bipartite graph such that $|U| = |V| = n$ and $\delta(G) > n/2$ (the least degree). Show that for any $2 \le k \le n$ and a matching m of size k there are at most n^2 matchings m' of size $k-1$ such that we can get from m' to m using an augmenting path of length at most 3.

   Solution: 7

Chapter 2

Theory

2.1 Probability 101

Probability 101

2.2 Markov Chain

Definition. *A* discrete-time Markov chain *is a sequence of random variables* $X_0, X_1, X_2, \dots$ *with the Markov property:*

$$\Pr[X_{n+1} = x \mid X_0 = x_0, X_1 = x_1, \dots, X_n = x_n] = \Pr[X_{n+1} = x \mid X_n = x_n]$$
$$\text{(if both are defined, i.e., } \Pr[X_0 = x_0, X_1 = x_1, \dots, X_n = x_n] > 0)$$

and the possible values of X_i form a countable set called the state space of the Markov chain.

The Markov property states that the process has no memory – the next state depends only on the current state. We will deal with a special case where the state space of each random variable will be the same and finite. Moreover we will deal with time-homogenous Markov chains, that is $\Pr[X_{n+1} \mid X_n] = \Pr[X_n \mid X_{n-1}]$ (the transition probabilities are time independent). Thus we will represent Markov chains by their transition matrices – if a Markov chain has n states its transition matrix is $P \in [0,1]^{n\times n}$ such that $P_{i,j} = \Pr[X_{n+1} = i \mid X_n = j]$ (thus column sums are equal to one).

If we take a probability distribution $\pi \in [0,1]^n$ and multiply it by the transition matrix we get the probability distribution after one step $P\pi$.

There are several interesting properties of Markov chains:

- We say that a MC is *irreducible* iff for each pair of states $i, j \in [n]$ there is a time $k \in \mathbb{N}$ such that $(P^k)_{i,j} > 0$ (we can get from any state to any state).
- We say that a MC is *periodic* iff there is a state $i \in [n]$ and a period $p \in \mathbb{N}, p > 1$ such that for each time $k \in \mathbb{N}$ we have $(P^k)_{i,i} > 0 \Rightarrow p \mid k$ that is probability of staying at state i is positive only for multiples of the period.
- We say that $\pi \in [0,1]^n$ is a *stationary distribution* of a given MC iff $P\pi = \pi$ (the distribution is the same after one step).

Theorem 1. *If a MC is aperiodic and irreducible it has a unique stationary distribution π. Moreover for all pairs of states $i, j \in [n]$ the we know that*

$$\lim_{t\to\infty} (P^t)_{i,j} = \pi_i$$

Chapter 3

Solutions

3.1 Tutorial

1.

- **Can you all hear me?**
- **If you are uncomfortable asking a question in English, just ask in Czech/Slovak and I will translate.**
- **Have you all taken:**
  - (a) **a probability course (discrete probability, random variables, expected value, variance, Markov, Chernoff)**
  - (b) **a linear algebra course (matrix operations, linear maps, eigenvectors and eivenvalues, discriminant)**
  - (c) **a graph theory course (what a combinatorial graph is, bipartite, complete, coloring)**
  - (d) **a combinatorics course (factorial, binomial coefficients)**
  - (e) **an algorithms / programming course (big-O notation, possibly understanding Python based on the other question)**
- **This class is heavy on theory. Are you interested in computer simulations and or implementations? If so:**
  - (a) **Python**
  - (b) **R**
  - (c) **C++**

2. **You are presented with two sealed envelopes. There are $k\$$ in one of those and $\ell\$$ in the other ($k, \ell \in \mathbb{N}$ but you do not know k, ℓ in advance). You may open an envelope and (based on what you see) decide to take this one or the other (without looking into both).**

   (a) **Is there a way how to walk away with the larger amoung of money with probability strictly larger than 0.5?**

   Solution: Pick an envelope uniformly at random. If you see $m\$$ toss a fair coin until you get Tails. If the number of tosses was strictly less than m keep the envelope, otherwise take the other. If $k < \ell$ then the probability of keeping the envelope with $k\$$ is strictly less than the probability of keeping the envelope with $\ell\$$.

   (b) **What is the expected value you walk away with (in terms of k, ℓ)?**

   Solution: Let us recall the sum of geometric series:

$$\begin{aligned} S &= \sum_{j=0}^{n} q^j \\ &= 1 + q + q^2 + \ldots + q^n \\ &= 1 + q\left(1 + q + q^2 + \ldots + q^{n-1}\right) \\ &= 1 + q(S - q^n) \end{aligned}$$

thus

$$\begin{aligned} S &= 1 + q(S - q^n) \\ S - qS &= 1 - q^{n+1} \\ S &= \frac{1 - q^{n+1}}{1 - q} && \text{(pokud } q \neq 1\text{)} \end{aligned}$$

and for the infinite case:

$$\begin{aligned} \sum_{j=0}^{\infty} q^j &= \lim_{n\to\infty} \sum_{j=0}^{n} q^j \\ &= \lim_{n\to\infty} \frac{1 - q^{n+1}}{1 - q} \\ &= \frac{1}{1 - q} && \text{(pokud } |q| < 1\text{)} \end{aligned}$$

Thus exactly n tosses have probability for the general case where Tails has probability p and Heads has probability $1 - p$:

$$\Pr[n \text{ tosses}] = (1 - p)^{n-1} p \qquad \text{(for any } n \in \mathbb{N}^+\text{)}$$

Probability of at most n tosses:

$$\begin{aligned} \Pr[1, 2, \ldots, n \text{ tosses}] &= \sum_{j=1}^{n} p(1 - p)^{j-1} \\ &= p \sum_{j=1}^{n} (1 - p)^{j-1} \\ &= p \frac{1 - (1 - p)^n}{1 - (1 - p)} \end{aligned}$$

$$= 1 - (1-p)^n$$

Probability that we keep $k\$$ (fair coin):

$$\begin{aligned}\Pr[\text{tosses } < k] &= \sum_{j=1}^{k-1} 0.5^j \\ &= 1 - 0.5^{k-1}\end{aligned}$$

Thus probability of walking away with $k\$$ is

$$\begin{aligned}\Pr[\text{winning } k\$] &= \frac{1}{2}(1-0.5^{k-1}) + \frac{1}{2}0.5^{\ell-1} \\ &= \frac{1}{2} - 0.5^k + 0.5^\ell \\ &= \frac{1}{2} + (0.5^\ell - 0.5^k)\end{aligned}$$

Thus the expected win is

$$\mathbb{E}[\text{win}] = k\left(\frac{1}{2} + (0.5^\ell - 0.5^k)\right) + \ell\left(\frac{1}{2} + (0.5^k - 0.5^\ell)\right)$$

(c) Simulate.

Solution:

```python
# https://docs.python.org/3/library/random.html
# Do not use for cryptography!
from random import randint
from random import random


def geometric(pr: float = 0.5) -> int:
    """pr is success probability, return the number of tosses until
    the first success."""
    assert pr > 0
    sample = 1
    fail_pr = 1 - pr
    while random() < fail_pr:
        sample += 1
    return sample


# Our unknown amounts.
envelopes = [5, 10]

N = 1000000        # Number of samples.
total_amount = 0   # Total sum that we got during all samples.
got_larger = 0     # Number of times we walked away with the larger sum.

for _ in range(N):
    # Pick the first envelope at random.
    chosen = randint(0, 1)
```

```
        if geometric() < envelopes[chosen]:
            # Keep this one.
            pass
        else:
            # Choose the other.
            chosen = 1 - chosen
        if envelopes[chosen] >= envelopes[1 - chosen]:
            got_larger += 1
        total_amount += envelopes[chosen]

k = envelopes[0]
l = envelopes[1]
pr_larger = 0.5 + abs(0.5**k - 0.5**l)
e_win = k * (0.5 + (0.5**l - 0.5**k)) + l * (0.5 + (0.5**k - 0.5**l))

print(f'Pr[selected larger] = {got_larger / N} (={pr_larger})')
print(f'E[win] = {total_amount / N} (={e_win})')

# Possible outcome:
# Pr[selected larger] = 0.529865 (=0.5302734375)
# E[win] = 7.649325 (=7.6513671875)
```

3. **Graph isomorphism. You have seen an interactive proof of graph non-isomorphism on the class. Can you come up with an interactive proof of graph isomorphism?**

   Solution:

   - Both the prover P and the verifier V know two graphs G_1, G_2.
   - The prover knows an isomorphism π such that $\pi(G_1) = G_2$. Formally $\pi\colon V(G_1) \to V(G_2)$ such that
   $$(u, v) \in E(G_1) \Leftrightarrow (\pi(u), \pi(v)) \in E(G_2).$$
   And by $\pi(G_1)$ we mean the graph $(\pi(V(G_1)), \{(\pi(u), \pi(v)) \mid (u, v) \in E(G_1)\})$.
   - For ease of presentation we set $V(G_1) = V(G_2) = [n] = \{1, 2, 3, \ldots, n\}$.
   - The prover picks uniformly random permutation $\sigma \in S_n$ and sends the graph $G = \sigma(G_1)$.
   - The verifier picks uniformly random number $i \in \{1, 2\}$ and asks verifier to present a permutation τ such that $\tau(G) = G_i$.
   - If $i = 1$ then the prover sends $\tau = \sigma^{-1}$. If $i = 2$ then the prover sends $\tau = (\sigma \circ \pi)^{-1}$.

   This is indeed an interactive proof:

   - If the prover knows the isomorphism π, then all answers are correct.
   - If G_1, G_2 are not isomorphic, then the verifier will pick a graph (either G_1 or G_2) which is not isomorphic with G with probability $1/2$.

   Again the prover learns nothing about the isomorphism. If you find these interactive proofs interesting, take a look at “Zero Knowledge Proofs”.

   Also note that our prover can be implemented efficiently as opposed to the case of graph non-isomorphism. In fact in some sense the prover proves that it knows the isomorphism (this can be made formal, see “Zero Knowledge Proofs of Knowledge”).

   It is natural to repeat this protocol more times in order to boost the probabilities. This is called probability amplification. We will investigate this much more during the semester.

4. **We will focus on random walks and their properties a lot.**

   (a) **Random walks are useful when analysing algorithms – "two coloring without monochromatic triangle" of three-colorable graph.**

   (b) **Random numbers in the computer are often expensive to generate, can we reduce number of used random bits (expanders)? Or even get a deterministic algorithm?**

   (c) **To sample from extremely large spaces.**

   Let $n \in \mathbb{N}$, say $n = 30$. Let us the following problem we start with $X_0 = \lfloor n/2 \rfloor$ and do the following process:

   - **if $X_i \in \{0, n\}$ we stop**
   - **we set $X_{i+1} = X_i + \delta$ where δ is picked uniformly at random from $\{-1, 1\}$**

   (a) **Is this a Markov chain (Definition 2.2)? If so can you write it's matrix?**

   Solution: Yes (see the lecture video).

   (b) **What is the expected number of steps until stopping?**

   Solution: Let us set

$$S_k = \mathbb{E}[\text{number of steps untill stopping, when starting at } k]$$

We know the following:

$$\begin{aligned} S_0 &= S_n = 0 \\ S_k &= 1 + \frac{1}{2}(S_{k-1} + S_{k+1}) && \text{(by linearity of expectation)} \end{aligned}$$

The above is so-called difference equation. It is not terribly complicated, but not super easy to solve (hint try to consider equations for $d(k) = S_k - S_{k-1}$ to get rid of the "1+" term). You may look at https://en.wikipedia.org/wiki/Recurrence_relation Luckily when dealing with asymptotics thus we do not need exact estimates. And you will see some nice theoretical results tomorrow.

But it can be shown that

$$S_k = k(n-k)$$

which we can easily check that this is indeed a solution (note that we would also need that this is a unique solution, see solution methods on Wikipedia for this part):

$$\begin{aligned} S_k &= 1 + \frac{1}{2}(S_{k-1} + S_{k+1}) \\ S_k &= 1 + \frac{1}{2}((k-1)(n-(k-1)) + (k+1)(n-(k+1))) \\ S_k &= 1 + \frac{1}{2}((k-1)n - (k-1)^2 + (k+1)n - (k+1)^2) \\ S_k &= 1 + \frac{1}{2}(2kn - (k-1)^2 - (k+1)^2) \\ S_k &= 1 + \frac{1}{2}(2kn - 2k^2 - 2) \\ S_k &= k(n-k) \end{aligned}$$

5. **Think of some example MCs.**

(a) Create a MC that is irreducible.

Solution: Two states:

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

(with probability 1/2 stay at the current state, with probability 1/2 switch to the other state).

(b) Create a MC that is not irreducible.

Solution: Two states:

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$$

(always stay at the first state or immediatelly go there).

(c) Create a MC that is periodic.

Solution: Three states:

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

(from the first state go always to the third, from the second always to the first and from the third always to the second).

(d) Create a MC that is not periodic.

Solution: Two states:

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

(with probability 1/2 stay at the current state, with probability 1/2 switch to the other state).

(e) Compute a stationary distribution of the following MC:

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

Solution: One eigenvalue is 1, the only stationary distribution $(1/2, 1/2)^T$. The other eigenvalue is 0 with the corresponding eigenvector $(1, -1)^T$ (this is not a distribution).

(f) Create a MC that has more stationary distributions.

Solution: Two states:

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

(always stay where we are).

6. **We are collectors and we want to collect all n kinds of coupons. Coupons are sold in packages which all look the same. Thus when we buy an coupon, we buy one of n kinds uniformly at random. This is known as the *coupon collector* problem.**

   (a) **What is the expected number of coupons we need to buy to get all kinds?**

   Solution: Let t_i be the time to collect the i-th coupon kind after we have collected $i-1$ coupons. The probability of buying the i-th coupon is

   $$\Pr[\text{getting } i\text{-th coupon when already having } i-1 \text{ coupons}] = \frac{n-(i-1)}{n}$$

   Thus t_i has geometric distribution (we are tossing the same probability and waiting for the first success). The expected value of t_i is:

   $$\mathbb{E}[t_i] = \frac{n}{n-(i-1)}$$

   By linearity of expectation:

   $$\begin{aligned}
   \mathbb{E}[\text{collecting}] &= \mathbb{E}[t_1 + t_2 + \ldots + t_n] \\
   &= \mathbb{E}[t_1] + \mathbb{E}[t_2] + \ldots + \mathbb{E}[t_n] \\
   &= \frac{n}{n} + \frac{n}{n-1} + \frac{n}{n-2} + \ldots + \frac{n}{n-(n-1)} \\
   &= nH_n \\
   &= n\log(n) + n \cdot 0.577\ldots + 1/2 + \mathcal{O}(1/n) && \text{(source Wikipedia)}
   \end{aligned}$$

   (b) **How many coupons do we need to buy to have probability at least $1-q$ of collecting all kinds?**

   Solution: We can use Markov inequality $\Pr[T > nH_n/q] \leq q$ (here T is the random variable telling us how many tosses are necessary).

   (c) **What is the Markov chain? Is this similar to a random walk on some graph?**

   Solution: There might be more Markov chains corresponding to this problem. The states could be all subsets of $[n] = \{1, 2, 3, \ldots, n\}$ (too big – not that nice to work with) or how many coupons have we collected so far (much smaller).

   This corresponds to the cover time of a complete graph (when we have loops in each vertes).

   (d) **Simulate.**

   Solution:

```python
import matplotlib.pyplot as plt
from collections import Counter
from random import randint

def catch_them_all(n: int = 50) -> int:
    coupons = [False] * n
    coupons_collected = 0
    coupons_bought = 0
    while coupons_collected < len(coupons):
        new_coupon = randint(0, len(coupons) - 1)
```

```
            coupons_bought += 1
            if not coupons[new_coupon]:
                coupons[new_coupon] = True
                coupons_collected += 1
    return coupons_bought

cnt = Counter(catch_them_all(50) for _ in range(10000))
plt.bar(cnt.keys(), cnt.values())
plt.xlabel("Steps untill collecting all 50 coupons")
plt.ylabel("How many times did we take this many steps")
# plt.show()
plt.savefig('coupon_collector.pdf')
```

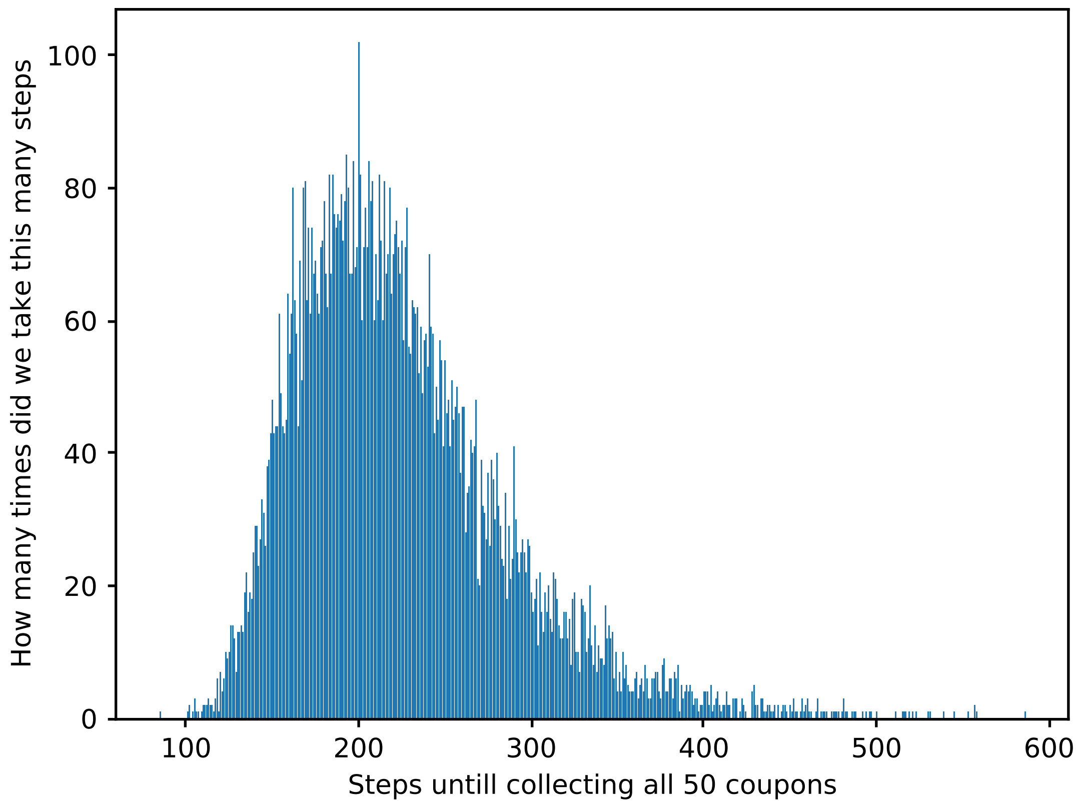


Figure 3.1: A histogram of how many steps were necessary (say 200 steps was necessary around 80 times).

3.2 Tutorial

1. **Find a family of oriented graphs of constant in-degree and constant out-degree and as large hitting time as possible.**

   Solution: Let us first do constant out-degree and unbounded in-degree. We will later use a tree to achieve constant in-degree.

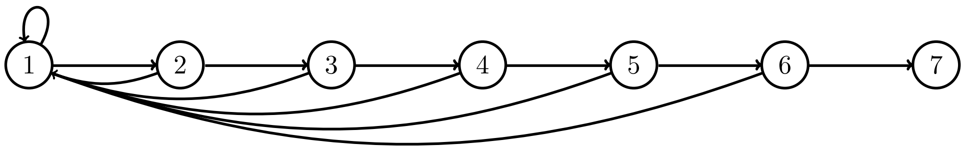


Figure 3.2: Oriented path with backwards arcs (oriented edges).

The expected hitting time from

$$h_{n,n} = 0$$

$$h_{n-1,n} = 1 + 0.5h_{1,n}$$

$$\begin{aligned} h_{n-2,n} &= 1 + 0.5(h_{1,n} + h_{n-1,n}) \\ &= 1 + 0.5(h_{1,n} + (1 + 0.5h_{1,n})) \\ &= 1.5 + 0.75h_{1,n} \end{aligned}$$

$$h_{n-3,n} = 1 + 0.5(h_{1,n} + (1 + 0.5(h_{1,n} + (1 + 0.5h_{1,n}))))$$

$$\begin{aligned} h_{n-k,n} &= \left(\sum_{j=0}^{k-1} 0.5^j\right) + \left(h_{1,n}\sum_{j=1}^{k} 0.5^j\right) \\ &= 2 - 2^{1-k} + h_{1,n}(1 - 2^{-k}) \end{aligned}$$

Thus in particular when $k = n - 1$:

$$\begin{aligned} h_{1,n} &= 2 - 2^{1-(n-1)} + h_{1,n}(1 - 2^{-(n-1)}) \\ 2^{-(n-1)}h_{1,n} &= 2 - 2^{1-(n-1)} \\ h_{1,n} &= 2^{n-1}(2 - 2^{1-(n-1)}) \\ &= 2^n - 2 \end{aligned}$$

Note that similar situation could happen on undirected graphs where the probabilities of traversing edge one way and the other way would not be the same. Which is in principle almost an oriented graph.

2. **Let $A \in \mathbb{R}^{n\times n}$ be a matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$. Show that the matrix $A + dI_n$ has eigenvalues $d + \lambda_1, \ldots, d + \lambda_n$.**

   Solution: Eigenvalues and eigenvectors recap:

   - We are interested in the limit of a Markov chain. When π_0 is the initial distribution, then $P^n\pi_0$ is the distribution after n steps.
   - When we are iteratively multiplying a vector by a matrix from left, the simplest form we can hope for are eigenvectors, which satisfy

     $$Ax = \lambda x$$

     Where A is a square matrix, λ is a real number called the *eigenvalue*, x is called the *eigenvector*. Then

     $$A^n x = A\left(A^{n-1}x\right) = \lambda^n x$$

   - For small matrices we usually use the characteristic polynomial:

     $$\det(A - \lambda I) = 0$$

     the roots of this polynomial are the eigenvalues and we find the corresponding eigenvectors as:

     $$A - \lambda I = \vec{0}$$

   - For an eigenvalue λ we define its *algebraic* multiplicity to be the multiplicity of λ as the root of the characteristic polynomial.
   - For an eigenvalue λ we define its *geometric* multiplicity to be the dimension of

     $$\mathrm{Ker}(A - \lambda I).$$

   - We know that for any eigenvalue algebraic multiplicity is at least the geometric multiplicity.
   - For each eigenvalue there is at least one eigenvector.
   - It is usually infeasible to find roots of the characteristic polynomial when the matrix A is large. There are however computationally efficient methods of computing eigenvalues and eigenvectors (usually iterative multiplication converges to the eigenvector).

   We use the definition, let λ be an eigenvalue in question and x its corresponding eigenvector:

   $$\begin{aligned} Ax &= \lambda x \\ (A + dI)x &= Ax + dIx \\ &= \lambda x + dx \\ &= (\lambda + d)x \end{aligned}$$

   Note that this can be rather useful when computing eigenvalues of a given matrix.

3. **Show Courant-Fisher: Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix ($A^T = A$). Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ be its eigenvalues. Show**

   (a) $\lambda_1 = \max_{x\in\mathbb{R}^n, \|x\|=1} x^T A x$

   Solution: Since A is Hermitian, we know that it is diagonalizable and we can choose an ortonormal basis of eigenvectors $u_1, u_2, \ldots, u_n$. That is for any j we have $u_j^T u_j = 1$ and $Au_j = \lambda_j u_j$, and for any $i \neq j$ we have $u_j^T u_i = 0$.

   We show two inequalities:

$$\begin{aligned}\max_{x\in\mathbb{R}^n, \|x\|=1} x^T A x &\geq u_j^T A u_j \\ &= u_j^T \lambda_j u_j \\ &= \lambda_j\end{aligned}$$

   On the other hand we may write $x = \alpha_1 u_1 + \alpha_2 u_2 + \ldots + \alpha_n u_n$ and thus get

$$\begin{aligned}\max_{x\in\mathbb{R}^n, \|x\|=1} x^T A x &= (\alpha_1 u_1 + \alpha_2 u_2 + \ldots + \alpha_n u_n)^T A (\alpha_1 u_1 + \alpha_2 u_2 + \ldots + \alpha_n u_n) \\ &= (\alpha_1 u_1 + \alpha_2 u_2 + \ldots + \alpha_n u_n)^T (\lambda_1 \alpha_1 u_1 + \lambda_2 \alpha_2 u_2 + \ldots + \lambda_n \alpha_n u_n) \\ &= \lambda_1 \alpha_1^2 + \lambda_2 \alpha_2^2 + \ldots + \lambda_n \alpha_n^2 \qquad \text{(since } u_j^T u_i = 0 \text{ and } u_j^T u_j = 1\text{)} \\ &\leq \lambda_1\end{aligned}$$

   Where the last equation follows from the fact that if Q is ortogonal matrix ($Q^T Q = I$) then $\|x\| = \|Qx\|$ since $\|x\| = x^T x$ and $\|Qx\| = x^T Q^T Q x$.

   (b) $\lambda_n = \min_{x\in\mathbb{R}^n, \|x\|=1} x^T A x$

   Solution: Consider $-A$ and use the previous result.

   (c) The eigenvalue λ_2 can be computed similarly $\lambda_2 = \max_{x\in\mathbb{R}^n, \|x\|=1, x^T u_1 = 0} x^T A x$ (where u_1 is the eigenvector corresponding to λ_1). We can get other eigenvalues in a similar manner. Moreover we could use this to prove the interlacing theorem. See https://en.wikipedia.org/wiki/Min-max_theorem

4. **Show that a connected d-regular graph is bipartite iff the least eigenvalue of its adjacency matrix is $-d$.**

 Solution: We know that the largest eigenvalue of the adjacency matrix of a d-regular graph is d and there is a corresponding eigenvector $(1, 1, \ldots, 1)^T$.

 If the graph is bipartite (that is $V(G) = A \cup B$ and $E(G) \subseteq A \times B$), we may use the vector defined as follows:

$$x_v = \begin{cases} -1 & \text{if } v \in A \\ 1 & \text{if } v \in B \end{cases}$$

 Then x is an eigenvector corresponding to $-d$.

 Let $(x_1, x_2, \ldots, x_n)^T$ be the eigenvector corresponding to $-d$. Thus

$$-dx_i = \sum_{j \in N(i)} x_j$$

 Let $M = \max_i |x_i|$ and $P = \{i \mid x_i = M\}$ and $N = \{i \mid x_i = -M\}$. Without loss of generality let P be non-empty. For any $i \in P$ we have

$$-dM = \sum_{j \in N(i)} x_j$$

 thus $x_j = -M$ for each $j \in N(i)$ (since each $|x_k| \leq M$).

 Since the graph is connected we eventually get that for any i it holds that $|x_i| = M$.

 Eigenvalues of the graph of neurons in human brain have been considered in epilepsy – they studied "how much" is the brain bipartite, which can be expressed by the difference between the smalles eigenvalue and the negative degree.

5. **Compute the eigenvalues and eigenvectors of the following graphs:**

   (a) K_n, the complete graph on n vertices.

   Solution: It will be easier to determine eigenvalues and eigenvectors of a complete graph with selfloops (we add unit matrix). We may subtract ones if we mind the selfloops.

   The adjacency matrix of a complete graph with selfloops is:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

   By the observation from the lecture we know that this is a regular graph and the matrix above has eigenvalue n with eigenvector $(1,1,1,1,1)^T$. By Hamiltonicity we know that all other eigenvectors are perpendicular to the one above. Thus all their entries sum up to zero.

   We guess other eigenvectors (we need to guess $n-1$ of them).

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

   Thus geometric (and thus also the algebraic) multiplicity of eigenvalue 0 is $n-1$.

   (b) $K_{n,n}$, the complete bipartite graph with partites of size n each.

   Solution: Here we are happy with no selfloops (otherwise the graph would not even be bipartite).

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

The graph is bipartite and regular, thus we know that the largest eigenvalue is n with the eigenvector $(1,1,1,1,1,1)^T$ the smallest eigenvalue is $-n$ with the eigenvector $(-1,-1,-1,1,1,1)^T$. As with the complete graph with selfloops it is easy to show that the rest is zero eigenvalues with corresponding vectors.

(c) C_n, the cycle on n vertices.

Solution: If we knew circular matrices we could use their properties. We will write the adjacency matrix as a sum of two simpler matrices:

$$\begin{pmatrix}0&1&0&0&0&1\\1&0&1&0&0&0\\0&1&0&1&0&0\\0&0&1&0&1&0\\0&0&0&1&0&1\\1&0&0&0&1&0\end{pmatrix} = \begin{pmatrix}0&0&0&0&0&1\\1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\end{pmatrix} + \begin{pmatrix}0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\1&0&0&0&0&0\end{pmatrix}$$

Observe that moreover the two matrices are inverse to each other, thus their eigenvalues are inverses as:

$$\begin{aligned} Ax &= \lambda x \\ x &= Ix \\ &= A^{-1}Ax \\ &= \lambda A^{-1}x \end{aligned}$$

Let $\omega \in \mathbb{C}$ be the primitive n-th root of unity. Thus $\omega = e^{2i\pi/n}$.

$$\begin{pmatrix}0&0&0&0&0&1\\1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\end{pmatrix}\begin{pmatrix}\omega^0\\\omega^1\\\omega^2\\\omega^3\\\omega^4\\\omega^5\end{pmatrix} = \begin{pmatrix}\omega^5\\\omega^0\\\omega^1\\\omega^2\\\omega^3\\\omega^4\end{pmatrix} = \omega^5\begin{pmatrix}\omega^0\\\omega^1\\\omega^2\\\omega^3\\\omega^4\\\omega^5\end{pmatrix}$$

Similarly for the even powers

$$\begin{pmatrix}0&0&0&0&0&1\\1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\end{pmatrix}\begin{pmatrix}\omega^0\\\omega^2\\\omega^4\\\omega^0\\\omega^2\\\omega^4\end{pmatrix} = \begin{pmatrix}\omega^4\\\omega^0\\\omega^2\\\omega^4\\\omega^0\\\omega^2\end{pmatrix} = \omega^4\begin{pmatrix}\omega^0\\\omega^2\\\omega^4\\\omega^0\\\omega^2\\\omega^4\end{pmatrix}$$

And so on. A particular eigenvector is an eigenvector of the eigenvalue ω^j with respect to this matrix and of eigenvalue ω^{-j} with respect to the inverse matrix. Thus when we sum the two matrices we get that the eigenvalue is $\omega^j + \omega^{-j}$. Observe that $\omega^j + \omega^{-j} \in \mathbb{R}$.

3.3 Tutorial

1. **You are given two coins. One is fair and the other one has $\Pr[\textbf{tails}] = 1/4$. We use the following algorithm to distinguish those:**

   - **Pick a coin and toss it n times.**
   - **Let $\hat{p}$ be the probability of getting a tails (number of tails over n).**
   - **If $\hat{p} \geq 3/8$ we say this coin is fair.**

   Show that if $n \geq 32\ln(2/\delta)$ then our algorithm answers correctly with probability at least $1-\delta$.

   Solution: Each coin is independent $0, 1$ random variable. We could have used the statement to get a similar bound:

$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/2}$$
$$\Pr[X \geq (1+\delta)\mu] \leq e^{-\delta^2\mu/3}$$

   but the following is a bit more convenient for us here:

$$\Pr[X \geq \mu + \delta n] \leq e^{-2n\delta^2}$$
$$\Pr[X \leq \mu - \delta n] \leq e^{-2n\delta^2}$$

   - If we were tossing the fair coin the probability of failure is

$$\begin{aligned} \mu &= n/2 \\ \delta &= 1/8 \\ \Pr[X \leq n/2 - n/8] &\leq e^{-2\cdot 32\ln(2/\delta)(1/8)^2} \\ \Pr[X \leq 3n/8] &\leq e^{-\ln(2/\delta)} \\ &\leq \delta/2 \end{aligned}$$

   - If we were tossing the tipped coin the probability of failure is

$$\begin{aligned} \mu &= n/4 \\ \delta &= 1/8 \\ \Pr[X \geq n/4 + n/8] &\leq e^{-2\cdot 32\ln(2/\delta)(1/8)^2} \\ \Pr[X \geq 3n/8] &\leq e^{-\ln(2/\delta)} \\ &\leq \delta/2 \end{aligned}$$

2. **You have seen that ZPP = RP $\cap$ co-RP.**

   (a) Recall definitions of:

   - **RP**

     Solution: A language $L \subseteq \{0,1\}^*$ is in RP ($L \in RP$) iff there is a probabilistic Turing machine A such that:

     – A works in polynomial time in the input length (that is $A(x)$ works in time $|x|$ for any $x \in \{0,1\}^*$).

     – If $x \notin L$ then $A(x) = 0$ always.

     – If $x \in L$ then $\Pr[A(x) = 1] \geq 1/2$ (the randomness is over the random bits of A).

   - **ZPP**

     Solution: A language $L \subseteq \{0,1\}^*$ is in RP ($L \in RP$) iff there is a probabilistic Turing machine A such that:

     – $A(x) = 1$ if and only if $x \in L$

     – A works in expected polynomial time (expectation is over the random bits of A).

   - **co-RP**

     Solution: L is in co-RP iff $\{0,1\}^* \setminus L$ is in RP.

   - **BPP**

     Solution: L is in BPP iff there is a probabilistic Turing machine A such that:

     – A works in polynomial time

     – If $x \in L$ then $\Pr[A(x) = 1] \geq 3/4$.

     – If $x \notin L$ then $\Pr[A(x) = 0] \geq 3/4$.

   - **NP**

     Solution: L is in NP if there is a deterministic Turing machine A such that:

     – A works in polynomial time in the input length (sum of input length and certificate length).

     – If $x \in L$ then there is $c \in \{0,1\}^*$ such that $|c|$ is polynomial in $|x|$ and $A(x, c) = 1$.

     – If $x \notin L$ then for any $c \in \{0,1\}^*$ we have $A(x, c) = 0$.

   (b) Show that RP $\subseteq$ NP (and thus co-RP $\subset$ co-NP).

   Solution: The random bits can serve as the certificate.

   (c) Decide if BPP = co-BPP.

   Solution: Yes, we can create B that on any x answers $1 - A(x)$.

   (d) Show that if NP $\subseteq$ BPP then NP=RP.

   Solution: We already know that RP $\subseteq$ NP (unconditionally), we thus need the other inclusion.

We know that 3SAT is NP-complete (if we can solve 3SAT, we can solve anything in NP). Thus it is enough to show that given A which is the BPP Turing machine for 3SAT we can do the following:

- If A rejects, reject.
- If A accepts, we hope the given formula is satisfiable and try to find an assignment:
  - Say the given formula φ has n variables.
  - If φ is satisfiable even if we set $x_1 = True$, we set it to True (otherwise to False).
  - We continue with $x_2, x_3, \ldots, x_n$.
  - Return $\varphi(x_1, x_2, \ldots, x_n)$.

  We need to be certain-enough when deciding the variables. Thus we run A multiple times – $\mathcal{O}(\log(n))$ times and take the majority answer to get probability $1 - 1/100n$ of correct answer. By union bound we get that probability of an error in any fixing of $x_1, x_2, \ldots, x_n$ is at most $1/100$.

  Determine the constant before the $\log(n)$ using a Chernoff bound.

3. **How to simulate a fair coin using a tipped coin and wice versa.**

   (a) **We are given a fair coin** $\Pr[tails] = 0.5$**. Show how to generate a random bit with** $\Pr[1] = p$ **for a given** $p \in (0, 1)$ **(both** $p = 0$ **and** $p = 1$ **are a bit boring).**

   Solution: Note that if the given p does not have finite binary representation there is no number T such that it would be enough to do at most T tosses. If at most T tosses would suffice, then imagine a tree of toss results. Any leaf is at depth at most T. In any leaf we output either 1 or 0. Probability of getting to a leaf is a multiple of 2^{-T} (not all leafs might be at the same depth).

   Say that $p = 0.p_1p_2p_3\ldots$ where p_j are binary digits. We treat the fair coin tosses as digits of a random number q. We toss until $q > p$ in which case we output 0 or we are sure that $q \leq p$ no matter the following tosses in which case we output 1.

   After each toss the probability of outputting is one half. Thus the expected number of tosses is constant.

   (b) **We are given a tipped coin – we do not even know** $p = \Pr[tails]$**. We are sure that** $\Pr[tails] \in (0, 1)$**. Generate a fair coin toss.**

   Solution: Algorithm:

   - We toss twice.
   - If the outcome was Heads, Tails we output 0.
   - If the outcome was Tails, Heads we output 1.
   - If the outcome was Heads, Heads or Tails, Tails we repeat.

   We know that:

   - Probability of outputting 0 is $p\,(1 - p)$.
   - Probability of outputting 1 is $(1 - p)\,p = p\,(1 - p)$.
   - Probability of outputting is $2p(1 - p) > 0$. Thus the expected number of rounds is $\frac{1}{2p(1-p)}$, which is finite for any $p \in (0, 1)$.

4. **Show that the expected number of comparisons a quick-sort algorithm does is roughly $n \ln(n)$. Show that probability of it making at least $32n \ln(n)$ comporisons is at most $1/n^3$.**

Solution: Our plan is to:

- Observe that if the total depth of recursion is k then the number of comparisons is upper bounded by kn (since each level of recursion causes at most n comparisons).
- Compute the probability that a fixed element is present in $> 32 \ln(n)$ levels of recursion.
- Use union bound to bound the probability there is an element which is present in $> 32 \ln(n)$ levels of recursion.

Let us do the second item.

- Let us fix an element s.
- Let $S_1 = n$, S_j be the size of the array containing s on the j-th level of recursion. Observe that at the end of recursion $S_k = 1$.
- We say that the j-th recursion is "lucky" if $S_{j+1} \leq (3/4)S_j$.
- Let us define an indicator variable X_j to denote if the j-th recursion is "lucky." Observe that $\Pr[X_j] = 1/2$ and X_i, X_j are independent for any $i \neq j$.
- After r lucky recursions in the first k levels we know that $S_k \leq (3/4)^r n$.
- So after $4 \ln(n) \geq \log_{3/4}(n)$ lucky recursions the element s is contained in an array of length one (and thus the recursion stops).
- Number of lucky rounds is equal to $X = \sum_{j=1}^{32 \ln(n)} X_j$. The expected number of lucky rounds is $\mu = 16 \ln(n)$. Let us set $\delta = 3/4$ and use Chernoff bound (independent indicator variables):

$$
\begin{aligned}
\Pr[X \leq (1-\delta)\mu] &\leq e^{-\delta^2 \mu/2} && \text{(the form we are using)} \\
\Pr[X \leq 4 \ln(n)] &\leq e^{-(3/4)^2 16 \ln(n)/2} \\
&\leq e^{-9 \ln(n)/2} \\
&\leq n^{-4}
\end{aligned}
$$

3.4 Tutorial

1. **We have k servers that are supposed to handle $n \gg k$ jobs. But the jobs come online and there is no single computer that knows the loads of servers (otherwise we would have a lot of communication). How do we distribute the jobs? We distribute the jobs each independently uniformly at random. How to bound the maximum load?**

   Solution:

   - Let X_i be the load of the i-th server.
   - We know that $X_i = \sum_{\ell=1}^{n} X_{i,\ell}$ where $X_{i,\ell}$ indicates if the ℓ-th job lands on the i-th server. And $X_{i,1}, X_{i,2}, \ldots, X_{i,n}$ are independent for each i (that does not hold for X_i).
   - $\Pr[X_{i,\ell}] = 1/k$
   - $\mathbb{E}[X_i] = n/k$
   - We use Chernoff bound:

   $$\Pr[X \geq (1+\delta)\mu] \leq e^{-\delta^2\mu/3}$$

   We set $\delta = 3\sqrt{k\ln(k)/n}$.

   $$\begin{aligned}
   \Pr[X_i \geq n/k + 3\sqrt{n\ln(k)/k})] &= \Pr[X_i \geq (1 + 3\sqrt{k\ln(k)/n})n/k] \\
   &\leq e^{-(3\sqrt{k\ln(k)/n})^2 n/3k} && \text{(Chernoff bound)} \\
   &\leq e^{-3\ln(k)} \\
   &= k^{-3}
   \end{aligned}$$

   - We use the union bound to bound the probability that there exists a server with that load:

   $$\Pr[\text{exists } i \in [k]\colon\ X_i \geq n/k + 3\sqrt{n\ln(k)/k})] = k^{-2}$$

   - To be concrete:

   $$\begin{aligned}
   k &= 1000 \\
   n &= 1000000 \\
   n/k &= 1000 \\
   \Pr[\text{exists a server with at least } n/k + 3\sqrt{n\ln(k)/k} \text{ jobs}] &\leq k^{-2} \\
   \Pr[\text{exists a server with at least 1250 jobs}] &\leq 1/1000000
   \end{aligned}$$

2. **Distributed discrete logarithm algorithm (Breaking the Circuit Size Barrier for Secure Computation Under DDH, Boyle, Gilboa, Ishai linked on the website).**

   Solution: We say that a group $\mathbb{G}$ is *cyclic* iff any of its element can be generated using a single generator. Say we have the group $(\mathbb{Z}_5^*, \cdot)$ with its generator 3:

$$
\begin{aligned}
&\mathbb{Z}_5^* = \{1, 2, 3, 4\} \\
&3^0 = 1 \\
&3^1 = 3 \\
&3^2 = 4 \\
&3^3 = 2
\end{aligned}
$$

   When we fix a group $\mathbb{G}$ and its generator g we may ask what is the discrete logarithm of a given group element:

$$DLog_{\mathbb{G},g}(x) = \min_{n \in \mathbb{N}} g^n = x$$

   So for instance $DLog_{\mathbb{Z}_5^*,3}(4) = 2$.

3. **Let A, B be two disjoint sets of vertices where $|A| = |B| = n$. Let $d \geq 5$ be a constant. We choose d uniformly at random edges from each vertex from A. We show that with constant positive probability each set $S \subseteq A$ of size $|S| \leq n/d$ has at least $\beta|S|$ neighbors where $\beta = d/4$.**

   Solution:

   - For each $S \subseteq A$ and for each $T \subseteq B$ we denote $X_{S,T}$ the indicator variable that is equal to one iff all the neighbors of S are contained in T.
   - $$\Pr[X_{S,T} = 1] = \left(\frac{|T|}{n}\right)^{d|S|}$$
   - We use the estimate that $\binom{n}{k} \leq \left(\frac{ne}{k}\right)^k$.
   - $$\begin{aligned}
\Pr[\exists S \subseteq A, T \subseteq B \colon |S| \leq n/d, |T| \leq \beta|S|, X_{S,T}] &\leq \sum_{s=1}^{n/d} \binom{n}{s}\binom{n}{\beta s}\left(\frac{\beta s}{n}\right)^{ds} \\
&\leq \sum_{s=1}^{n/d} \binom{n}{\beta s}^2 \left(\frac{\beta s}{n}\right)^{ds} \\
&\qquad \left(\text{as } \tbinom{n}{s} \leq \tbinom{n}{\beta s}\right) \\
&\leq \sum_{s=1}^{n/d} \left(\frac{ne}{\beta s}\right)^{2\beta s} \left(\frac{\beta s}{n}\right)^{ds} \\
&= \sum_{s=1}^{n/d} \left(\frac{4ne}{ds}\right)^{ds/2} \left(\frac{ds}{4n}\right)^{ds} \\
&= \sum_{s=1}^{n/d} \left(\frac{eds}{4n}\right)^{ds/2} \\
&\leq \sum_{s=1}^{n/d} \left(\frac{e}{4}\right)^{ds/2} \qquad (\text{as } ds \leq n) \\
&\leq \frac{(e/4)^{d/2}}{1-(e/4)^{d/2}} \qquad (\text{geometric series}) \\
&< 1
\end{aligned}$$
   - What would happen if the graph was a union of d perfect matchings?

3.5 Tutorial

1. **Let us define the *edge expansion* for a given graph G by:**

$$h(G) = \min_{|S| \leq n/2} \frac{e(S, V \setminus S)}{|S|}$$

For any $S \subseteq V(G)$ we denote

$$e(S) = E(G) \cap S \times S = \textbf{number of edges inside } S$$
$$e(S, V(G) \setminus S) = E(G) \cap (S \times (V(G) \setminus S)) = \textbf{number of edges going from } S \textbf{ to the complement}$$

Let us show that if λ_2 is the second largest eigenvalue of the adjacency matrix of a d-regular graph G then:

$$h(G) \geq \frac{d - \lambda_2}{2}$$

Solution:

- The idea is to use Courant-Fisher of Problem 3 from the second tutorial. If u_1 is the eigenvector corresponding to the first eigenvalue λ_1, we have:

$$\lambda_2 = \max_{x \in \mathbb{R}^n, x^T u_1 = 0} \frac{x^T A x}{x^T x}$$

- Recall that $u_1 = (1, 1, 1, \ldots, 1)^T$ and our vector x should be orthogonal to it so that we can use the Courant-Fisher ($\langle x \mid u_1 \rangle = 0$). Moreover it should correspond to our set S.
- Let $S \subseteq V(G)$ of size $s = |S| \leq n/2$. Let us define the vector

$$x_v = \begin{cases} n - s & v \in S \\ -s & v \notin S \end{cases}$$

- Let us determine the norm squared of x:

$$\begin{aligned} x^T x &= x^T x \\ &= (n-s)^2 s + s^2 (n-s) \\ &= s(n-s)n \end{aligned}$$

- Let us determine the nominator from Courant-Fischer for our vector x as defined above:

$$\begin{aligned} x^T A x &= 2 \sum_{(u,v) \in E(G)} x_u x_v \\ &= 2(n-s)^2 e(S) - 2s(n-s) e(S, V(G) \setminus S) + 2s^2 e(V(G) \setminus S) \end{aligned}$$

  Note that:

$$\begin{aligned} ds &= 2e(S) + e(S, V(G) \setminus S) \\ d(n-s) &= 2e(V(G) \setminus S) + e(S, V(G) \setminus S) \end{aligned}$$

we plug that into the above:

$$\begin{aligned}
x^T Ax &= 2 \sum_{(u,v)\in E(G)} x_u x_v \\
&= 2(n-s)^2 e(S) - 2s(n-s)e(S, V(G)\setminus S) + 2s^2 e(V(G)\setminus S) \\
&= (n-s)^2 \left(ds - e(S, V(G)\setminus S)\right) - 2s(n-s)e(S, V(G)\setminus S) + s^2 \left(d(n-s) - e(S, V(G)\setminus S)\right) \\
&= e(S, V(G)\setminus S)\left(-(n-s)^2 - 2s(n-s) - s^2\right) + ds\left((n-s)^2 + s(n-s)\right) \\
&= -n^2 e(S, V(G)\setminus S) + dsn(n-s)
\end{aligned}$$

and we plug our vector x into the Courant-Fisher:

$$\begin{aligned}
\lambda_2 &\geq \frac{x^T Ax}{x^T x} \\
&= \frac{-n^2 e(S, V(G)\setminus S) + dsn(n-s)}{s(n-s)n} \\
&= d - e(S, V(G)\setminus S)\frac{n}{s(n-s)}
\end{aligned}$$

Finally we use that $s \leq n/2$ and thus $\frac{n-s}{n} \geq 1/2$ and rearrange the former inequality:

$$\begin{aligned}
\frac{e(S, V(G)\setminus S)}{|S|} &\geq \frac{n-s}{n}(d - \lambda_2) \\
&\geq \frac{d - \lambda_2}{2}
\end{aligned}$$

2. **Show that for any $v \in \mathbb{R}^n$ it holds that**

$$\frac{1}{\sqrt{n}}\|v\|_1 \leq \|v\|_2 \leq \|v\|_1$$

Solution: We use the Cauchy-Schwarz inequality:

$$\langle u \mid v\rangle \leq \|u\|_2\|v\|_2 \qquad \text{(for any } u, v \text{ and norm } \|v\|_2 = \sqrt{\langle v \mid v\rangle}\text{)}$$

The first inequality can be done by a clever choice of u:

$$\begin{aligned} u_i &= \begin{cases} 1 & \text{if } v_i \geq 0 \\ -1 & \text{if } v_i < 0 \end{cases} \\ \|v\|_2 &\geq \frac{\langle u \mid v\rangle}{\|u\|_2} \\ &= \frac{\sum_{i=1}^n |v_i|}{\sqrt{\sum_{i=1}^n u_i^2}} \\ &= \frac{\|v\|_1}{\sqrt{n}} \end{aligned}$$

The second inequality can be proven as follows:

$$\begin{aligned} \|v\|_1^2 &= \left(\sum_{i=1}^n |v_i|\right)\left(\sum_{i=1}^n |v_i|\right) \\ &= \left(\sum_{i=1}^n |v_i|^2\right) + \left(\sum_{i=1}^n \sum_{j\neq i} |v_i||v_j|\right) \\ &\geq \left(\sum_{i=1}^n |v_i|^2\right) \\ &= \|v\|_2^2 \end{aligned}$$

Let us just note that inequalities with norms are very useful and there are many of those. One well known is for instance the Hölder inequality: https://en.wikipedia.org/wiki/H%C3%B6lder%27s_inequality

3. **Let μ be a probability distribution, that is $\|\mu\|_1 = 1$ and $\mu_j \geq 0$ (for each $j \in \Omega$). Let us define $d(\mu, \nu)$ the distance of two probability distributions as:**

$$d(\mu, \nu) = \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)|$$

Show that:

$$d(\mu, \nu) = \max_{A \subseteq \Omega} \mu(A) - \nu(A)$$

where $\mu(A) = \sum_{x \in A} \mu(x)$.

Solution: Set the set $A = \{x \in \Omega \mid \mu(x) \geq \nu(x)\}$ and we get:

$$\begin{aligned}
\frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)| &= \frac{1}{2} \left(\sum_{x \in A} |\mu(x) - \nu(x)| \right) + \frac{1}{2} \left(\sum_{x \in \Omega \setminus A} |\mu(x) - \nu(x)| \right) \\
&= \frac{1}{2} \left(\sum_{x \in A} \mu(x) - \nu(x) \right) + \frac{1}{2} \left(\sum_{x \in \Omega \setminus A} \nu(x) - \mu(x) \right) \\
&= \frac{1}{2} \left(\mu(A) - \nu(A) \right) + \frac{1}{2} \left(\nu(\Omega \setminus A) - \mu(\Omega \setminus A) \right) \\
&= \frac{1}{2} \left(\mu(A) - \nu(A) \right) + \frac{1}{2} \left((1 - \nu(A)) - (1 - \mu(A)) \right) \\
&= \mu(A) - \nu(A) \\
&\leq \max_{A \subseteq \Omega} \mu(A) - \nu(A)
\end{aligned}$$

We are left to realize that our set A maximizes the right hand side.

4. **Let M be a Markov chain on the set of states S. We say that a Markov chain $Z_t = (X_t, Y_t)$ on the set of states $S \times S$ is a coupling iff**

$$\Pr[X_{t+1} = x' \mid Z_t = (x,y)] = \Pr[M_{t+1} = x' \mid M_t = x]$$
$$\text{(where } X_{t+1} \text{ is the first coordinate of } Z_{t+1})$$
$$\Pr[Y_{t+1} = y' \mid Z_t = (x,y)] = \Pr[M_{t+1} = y' \mid M_t = y]$$
$$\text{(where } Y_{t+1} \text{ is the second coordinate of } Z_{t+1})$$

So one can imagine a coupling as a Markov chain, that in both coordinates behaves in the same way as the original Markov chain (but the coordinates might be dependent on each other).

Let $Z_t = (X_t, Y_t)$ be a coupling of a Markov chain M on the state space S. Suppose there is a T such that:

$$\Pr[X_T \neq Y_T \mid X_0 = x, Y_0 = y] \leq \varepsilon \qquad \text{(for all } x, y \in S)$$

then

$$\tau(\varepsilon) \leq T$$

Where formally the mixing time $\tau(\varepsilon)$ is defined as

$$p_x^t = \textbf{ the distribution when starting at } x \textbf{ and doing } t \textbf{ steps}$$
$$\tau(\varepsilon) = \max_{x \in S} \min \left\{ t \mid d(p_x^t, \pi) \leq \varepsilon \right\}$$

Notice that when we prove that $\Pr[X_T \neq Y_T \mid X_0 = x, Y_0 = y] \leq \varepsilon$ for all $x, y \in S$, we know that we are close to the stationary distribution (without even knowing the stationary distribution).

Solution: Pick any set of states $A \subseteq S$ and try to bound the probability that after T steps X_T is in A and let Y_0 be selected accoding to the stationary distribution π:

$$\begin{aligned}
\Pr[X_T \in A] &\geq \Pr[X_T \in A \wedge X_T = Y_T] \\
&= \Pr[X_T = Y_T \wedge Y_T \in A] \\
&= 1 - \Pr[X_T \neq Y_T \vee Y_T \notin A] && \text{(probability of complement)} \\
&\geq 1 - \Pr[X_T \neq Y_T] - \Pr[Y_T \notin A] && \text{(union bound)} \\
&= (1 - \Pr[Y_T \notin A]) - \Pr[X_T \neq Y_T] \\
&\geq (1 - \Pr[Y_T \notin A]) - \varepsilon && \text{(assumption)} \\
&= \Pr[Y_T \in A] - \varepsilon && \text{(probability of complement)} \\
&= \pi(A) - \varepsilon
\end{aligned}$$

So for any $A \subseteq S$ we have

$$\Pr[X_T \in A] \geq \pi(A) - \varepsilon$$

and the same argument for $S \setminus A$ gives us

$$\Pr[X_T \notin A] \geq \pi(S \setminus A) - \varepsilon$$

so

$$\begin{aligned}
\Pr[X_T \notin A] &\geq \pi(S \setminus A) - \varepsilon \\
1 - \Pr[X_T \in A] &\geq 1 - \pi(A) - \varepsilon \\
-\Pr[X_T \in A] &\geq -\pi(A) - \varepsilon
\end{aligned}$$

$$\Pr[X_T \in A] \leq \pi(A) + \varepsilon$$

Thus together we have:

$$\pi(A) - \varepsilon \leq \Pr[X_T \in A] \leq \pi(A) + \varepsilon$$

so

$$\max_{x \in S} d(p_x^T, \pi) = \max_{x \in S, A \subseteq S} |p_x^T(A) - \pi(A)| \leq \varepsilon$$

5. **We define the hypercube graph of dimension d as follows: the vertices are binary strings of length d and two vertices are connected by an edge iff they differ in exactly one coordinate. For instance $d = 2$ the graph is**

$$(\{00, 01, 10, 11\}, \{(00, 01), (00, 10), (11, 01), (11, 10)\})$$

(the edges are not oriented).

We start at O^d and do the following random walk:

- **With probability 1/2 we stay at the current vertex.**
- **With probability 1/2 we choose uniformly at random and index $j \in [d]$ and change the corresponding bit.**

The Markov chain is nice (it converges to a single stacionary distribution, namely the uniform distribution on all vertices). Our question is how many steps do we need to take until we are "close enough" to the uniform distribution. Show that the random walk has $\tau(\varepsilon) \leq d \ln(d/\varepsilon)$.

Solution: We do coupling (X_t, Y_t) where

- $X_0 = O^d$
- Y_0 is chosen according to the uniform distribution (that is the stationary distribution)
- $Z_t = (X_t, Y_t)$ where the moves are as follows, let $X_t = x \in \{0,1\}^d$ and $Y_t = y \in \{0,1\}^d$ (that is $Z_t = (x, y)$):
  - Pick uniformly at random a coordinate $i \in [d]$ (the same for both)
  - If $x_i = y_i$ then with probability 1/2 we keep the i-th bit the same $Z_{t+1} = (X_{t+1}, Y_{t+1}) = (x, y)$ and with probability 1/2 we change it $Z_{t+1} = (x \oplus e_i, y \oplus e_i)$.
  - If $x_i \neq y_i$ then with probability 1/2 we keep the i-th bit of x the same and change the i-th bit of y $Z_{t+1} = (X_{t+1}, Y_{t+1}) = (x, y \oplus e_i)$ and with probability 1/2 we change the i-th bit of x and keep the i-th bit of y $Z_{t+1} = (x \oplus e_i, y)$.

Thus after picking the coordinate j we know for sure that $(X_t)_j = (Y_t)_j$ and it stays the same (the j-th bit is the same from that point on). Thus we have coupon collector problem. Probability of not picking all coordinates after $d \ln(d/\varepsilon)$ can be bounded by:

$$\begin{aligned}
\Pr[\text{there is a coordinate that has not been picked}] &\leq d \cdot \Pr[\text{coordinate } d \text{ has not been picked}] && \text{(union bound)} \\
&\leq d\,(1 - 1/d)^{d \ln(d/\varepsilon)} \\
&\leq d e^{-\ln(d/\varepsilon)} && (1 - x \leq e^{-x}) \\
&\leq \varepsilon
\end{aligned}$$

SILLY EX.: estimating π

sample points unif at rand

$G = \bigcirc$ $U = \square$

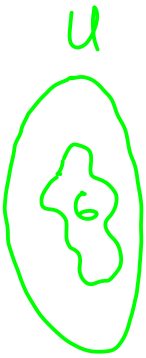
$3 \leq \pi \leq 4$

$\leadsto$ n can be deduced

GIVEN: $|U|$ ... usually triv
ε, δ ... parameters of the output
Q: $|G|$ ... we can usually guess $? \leq |G| \leq ?'$

3.6 Tutorial

output of rand. alg.



1. **Definition: a random variable – our estimate $A > 0$ is an $\varepsilon - \delta$ approximation of a value $g > 0$ if**

$|U| = \dots$ ✓
$|G| = ?$

$$\Pr\left[(1-\varepsilon)g \leq A \leq (1+\varepsilon)g\right] \geq 1-\delta$$

Prove the *Estimator Theorem*: Let U be a finite set and $G \subseteq U$ its subset. We know $|U|$ and wish to estimate $|G|$. If we take n uniformly random and independent samples from U where

$$\boxed{n \geq \frac{3}{\varepsilon^2 \frac{|G|}{|U|}} \ln(2/\delta)}$$

$X = \sum_{j=1}^{m} X_j \qquad 0 \leq X_j \leq 1$

$|G| \doteq A = X \cdot \frac{|U|}{m}$

$X =$ **number of samples inside of G**

$X_j = [j\text{-th sample} \in G]$

$\mathbb{E}[X] = ?$

and output $A = X\frac{|U|}{n}$ then A is $\varepsilon - \delta$ approximation of $|G|$.

Solution: We will use a Chernoff bound.

- Observe that $\mathbb{E}[X/n] = |G|/|U|$ as

$$X = \sum_{j=1}^{n} X_j \qquad (X_j \text{ is the indicator if } j\text{-th sample was in } G)$$

$$\boxed{\Pr[X_j] = \frac{|G|}{|U|}}$$

$$\boxed{\mathbb{E}[X_j] = \frac{|G|}{|U|}}$$

$$\boxed{\mathbb{E}[X] = n\frac{|G|}{|U|}} \qquad \text{(linearity of } \mathbb{E})$$

$$\mathbb{E}[A] = |G|$$

- All of X_j are independent and $0 \leq X_j \leq 1$.
- We may use Chernoff bounds, we use the following form:

$$\Pr[X \geq (1+\delta)\mu] \leq e^{-\delta^2\mu/3}$$

$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/2} \leq \boxed{e^{-\delta^2\mu/3}}$$

$$\Pr[X \geq (1+\delta)\mu \text{ or } X \leq (1-\delta)\mu] \leq 2e^{-\delta^2\mu/3}$$

- Plugging all of the above:

$$\Pr\left[X \geq (1+\varepsilon)\underbrace{n\frac{|G|}{|U|}}_{\mu} \text{ or } X \leq (1-\varepsilon)n\frac{|G|}{|U|}\right] \leq 2e^{-\varepsilon^2 n \frac{|G|}{3|U|}}$$

$$\leq 2e^{-\cancel{\varepsilon^2}\left(\frac{3}{\cancel{\varepsilon^2}\frac{|G|}{|U|}}\ln(2/\delta)\right)\frac{|G|}{3|U|}}$$

$$\leq 2e^{-(\cancel{3}\ln(2/\delta))/\cancel{3}}$$

$$= \delta$$

ε-APX

$\hat{s} \doteq s$

$\hat{t} \doteq t$

HOW GOOD APX.?

$\frac{\hat{s}}{\hat{t}} \doteq \frac{s}{t}$

4ε-APX

2. **We say that $\hat{x}$ is an ε-aproximation of x iff** >0

$$(1-\varepsilon)x \leq \hat{x} \leq (1+\varepsilon)x$$

Show that for $\varepsilon < 1/2$ if we have ε-aproximation $\hat{s}$ of a number s and ε-aproximation $\hat{t}$ of a number t then $\hat{s}/\hat{t}$ is an 4ε-aproximation of s/t.

Solution: We have

$$(1-\varepsilon)s \leq \hat{s} \leq (1+\varepsilon)s$$

$$(1-\varepsilon)t \leq \hat{t} \leq (1+\varepsilon)t$$

thus

$$\frac{\hat{s}}{\hat{t}} \leq \frac{(1+\varepsilon)s}{(1-\varepsilon)t}$$

and we need the following inequality to hold

$$\begin{aligned}
\boxed{\frac{1+\varepsilon}{1-\varepsilon} \leq 1+4\varepsilon} & \\
1+\varepsilon \leq (1+4\varepsilon)(1-\varepsilon) & \qquad (1-\varepsilon > 0) \\
1+\varepsilon \leq 1+3\varepsilon-4\varepsilon^2 & \\
0 \leq 2\varepsilon-4\varepsilon^2 &
\end{aligned}$$

$\varepsilon < \frac{1}{2}$

roots $0, \frac{1}{2}$

which holds for any $\varepsilon \in (0, 1/2)$.

The other inequality follows the same way:

$$\begin{aligned}
\frac{1-\varepsilon}{1+\varepsilon} \geq 1-4\varepsilon & \\
1-\varepsilon \geq (1-4\varepsilon)(1+\varepsilon) & \qquad (1+\varepsilon > 0) \\
1-\varepsilon \geq 1-3\varepsilon-4\varepsilon^2 & \\
0 \geq -2\varepsilon-4\varepsilon^2 &
\end{aligned}$$

which holds for any $\varepsilon \in (0, 1/2)$.

$\hat{a}_j$ is $\bar{\varepsilon}$-apx of a_j

we wish to ε-apx $\prod a_j$? how to set $\bar{\varepsilon}$

$a_j > 0$

3. **Let $\varepsilon > 0$ be fixed. Find a suitable choice of $\bar{\varepsilon}$ such that if we take $(\hat{a}_i)_{i=1}^n$ of numbers $(a_i)_{i=1}^n$ then $\prod_{i=1}^n \hat{a}_i$ is an ε-aproximation of $\prod_{i=1}^n a_i$.**

Solution: We know that

$$\begin{aligned}\prod_{i=1}^n \hat{a}_i &\le \prod_{i=1}^n (1+\bar{\varepsilon})a_i \\ &\le \left(\prod_{i=1}^n (1+\bar{\varepsilon})\right)\left(\prod_{i=1}^n a_i\right) \\ \text{WISH } &\le (1+\varepsilon)\left(\prod_{i=1}^n a_i\right)\end{aligned}$$

Thus we want

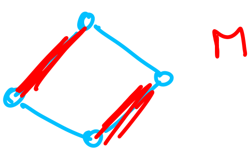
$$\begin{aligned}(1+\bar{\varepsilon})^n &\le 1+\varepsilon \\ 1+\bar{\varepsilon} &\le \sqrt[n]{1+\varepsilon} \\ \bar{\varepsilon} &\le \sqrt[n]{1+\varepsilon} - 1\end{aligned}$$

The same way we want from below

$$\begin{aligned}(1-\bar{\varepsilon})^n &\ge 1-\varepsilon \\ 1-\bar{\varepsilon} &\ge \sqrt[n]{1-\varepsilon} \\ -1+\bar{\varepsilon} &\le -\sqrt[n]{1-\varepsilon} \\ \bar{\varepsilon} &\le 1-\sqrt[n]{1-\varepsilon}\end{aligned}$$

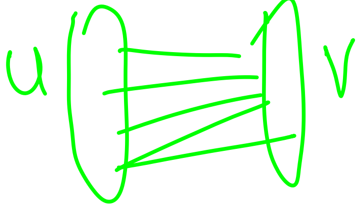
$\bar{\varepsilon} = \min\left(\sqrt[n]{1+\varepsilon}-1,\ 1-\sqrt[n]{1-\varepsilon}\right)$

$M \subseteq E(G)$ is a matching if each vertex of $V(G)$ has ≤ 1 edge from M



M is perfect if $|M| = \frac{|V(G)|}{2}$

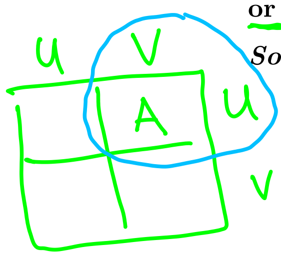
(or $= 1$ edge from M)



$|U| = |V|$

4. **Show an algorithm that given a bipartite graph G (partites consisting of the same number of vertices) determines if the number of perfect matchings is even or odd.**

?



Solution: Let us recall the definitions:

we know how to compute poly alg. $\mathcal{O}(n^3)$... Gaussian elim.

$$\det(A) = \sum_{\pi \in S_n} \underbrace{\operatorname{sgn}(\pi)}_{\{\pm 1\}} \prod_{i=1}^{n} A_{i,\pi(i)}$$

$$\operatorname{perm}(A) = \sum_{\pi \in S_n} \prod_{i=1}^{n} A_{i,\pi(i)} \qquad i \in V,\ \pi(i) \in U$$

$|V| = |U| = n$

$U \neq V$

permanent is thought to be hard

If A is the part of the adjacency matrix of G corresponding to the different partites (thus rows of A correspond to one partite and columns to the other and $A_{u,v} = 1$ iff u belongs to one partite, v to the other one and are connected together by an edge) and a permutation π determines a perfect matching (that is $(j, \pi(j)) \in E(G)$) then the product is equal to one (and to zero otherwise). Thus $\operatorname{perm}(A)$ is the number of perfect matchings. But remember that over $\mathbb{Z}_2$ we have $1 = -1$, thus specially $\det(A) = \operatorname{perm}(A)$ over $\mathbb{Z}_2$.

We know how to compute determinant in polynomial time. Permanent is thought to be hard to compute, but its parity is easy.

$\mathbb{Z}_2 = \{0, 1\}$ $\quad 1 + 1 \equiv_2 0 \quad 1 \equiv_2 -1$

physics cares about # of matchings especially on planar graphs (or on bounded genus)

g of plane is 2 ?

↳ g ... genus $\Rightarrow$ perm is a linear combination of 4^g determinants

Kastellein (or just ask prof Loebl)

$\boxed{\mathbb{E}[\det(B)^2] = \text{perm}(A)}$

over choice (ε_{ij})

$A = \begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix} \rightsquigarrow B = \begin{pmatrix}-1 & 1\\ 0 & -1\end{pmatrix}$

5. **Let $A \in \{0,1\}^{n\times n}$ be a matrix. Let $\varepsilon_{i,j}$ be independent random ± 1 variables. Let $B \in \{-1,0,1\}^{n\times n}$ be a matrix such that $B_{i,j} = \varepsilon_{i,j}A_{i,j}$ (uniformly randomly independently assign signs to entries of A).**

   (a) Show that $\mathbb{E}[\det(B)] = 0$

   Solution: Let us remind that

$$\det(A) = \sum_{\pi\in S_n} \text{sgn}(\pi) \prod_{i=1}^{n} A_{i,\pi(i)}$$

$$\text{perm}(A) = \sum_{\pi\in S_n} \prod_{i=1}^{n} A_{i,\pi(i)}$$

We use linearity of expectation (it holds even when the variables are dependent):

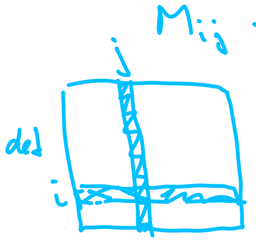
$$\begin{aligned}
\mathbb{E}[\det(B)] &= \mathbb{E}\left[\sum_{\pi\in S_n} \text{sgn}(\pi) \prod_{i=1}^{n} B_{i,\pi(i)}\right] \\
&= \sum_{\pi\in S_n} \text{sgn}(\pi)\mathbb{E}\left[\prod_{i=1}^{n} B_{i,\pi(i)}\right] \\
&= \sum_{\pi\in S_n} \text{sgn}(\pi)\mathbb{E}\left[\prod_{i=1}^{n} \varepsilon_{i,\pi(i)}A_{i,\pi(i)}\right] \\
&= \sum_{\pi\in S_n} \text{sgn}(\pi)0
\end{aligned}$$

linearity of $\mathbb{E}$

$= \sum_{\pi\in S_n} \text{sgn}(\pi) \prod_{i=1}^{n} \mathbb{E}[\varepsilon_{i,\pi(i)} A_{i,\pi(i)}]$

as $\varepsilon_{i,\pi(i)} A_{i,\pi(i)}$ and $\varepsilon_{j,\pi(j)} A_{j,\pi(j)}$ are independent

$M_{ij} =$



Here we could have used that for *independent* variables we have $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$, but I believe the above is clear enough.

We could have probably used the Laplace expansion

$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j} A_{i,j} M_{i,j} \qquad \text{(for any } i\text{)}$$

$M_{i,j} =$ determinant of the matrix A without i-th row and without j-th column

$$\text{perm}(A) = \sum_{j=1}^{n} A_{i,j} N_{i,j} \qquad \text{(for any } i\text{)}$$

$N_{i,j} =$ permanent of the matrix A without i-th row and without j-th column

   (b) Show that $\mathbb{E}[\det(B)^2] = \text{perm}(A)$ (permanent of A)

   Solution: We could investigate two directions

$$\det(B)^2 = \det(B^2)$$

let us go with the first one:

$$\begin{aligned}
\mathbb{E}[\det(B)^2] &= \mathbb{E}\left[\left(\sum_{\pi\in S_n} \text{sgn}(\pi) \prod_{i=1}^{n} B_{i,\pi(i)}\right)\left(\sum_{\sigma\in S_n} \text{sgn}(\sigma) \prod_{i=1}^{n} B_{i,\sigma(i)}\right)\right] \\
&= \mathbb{E}\left[\sum_{(\pi,\sigma)\in S_n\times S_n} \text{sgn}(\pi)\,\text{sgn}(\sigma) \prod_{i=1}^{n} B_{i,\pi(i)}B_{i,\sigma(i)}\right]
\end{aligned}$$

linearity $\qquad\qquad\qquad\qquad \mathrm{sgn}(\pi)^2 = 1$

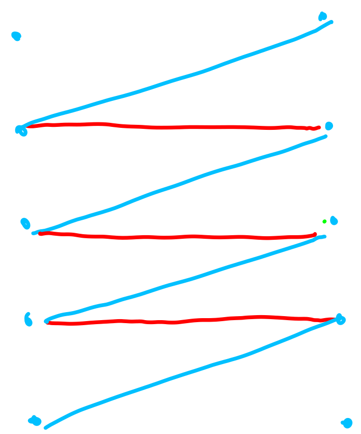
$$
\begin{aligned}
&= \mathbb{E}\left[\sum_{\pi\neq\sigma\in S_n} \mathrm{sgn}(\pi)\,\mathrm{sgn}(\sigma)\prod_{i=1}^{n} B_{i,\pi(i)}B_{i,\sigma(i)}\right] + \mathbb{E}\left[\sum_{\pi\in S_n}\prod_{i=1}^{n} B_{i,\pi(i)}^2\right] \\
&= \sum_{\pi\neq\sigma\in S_n} \mathrm{sgn}(\pi)\,\mathrm{sgn}(\sigma)\mathbb{E}\left[\prod_{i=1}^{n} B_{i,\pi(i)}B_{i,\sigma(i)}\right] + \underbrace{\mathbb{E}\left[\sum_{\pi\in S_n}\prod_{i=1}^{n} \underbrace{A_{i,\pi(i)}^2}_{\{0,1\}}\right]}_{} \qquad \varepsilon^2 = 1 \\
&= \sum_{\pi\neq\sigma\in S_n} \mathrm{sgn}(\pi)\,\mathrm{sgn}(\sigma)\mathbb{E}\left[\prod_{i=1}^{n} \varepsilon_{i,\pi(i)}\varepsilon_{i,\sigma(i)}A_{i,\pi(i)}A_{i,\sigma(i)}\right] + \mathrm{perm}(A) \qquad \exists j \\
&= \sum_{\pi\neq\sigma\in S_n} \mathrm{sgn}(\pi)\,\mathrm{sgn}(\sigma)0 + \mathrm{perm}(A) \\
&= \mathrm{perm}(A)
\end{aligned}
$$

$$\mathrm{perm}\begin{pmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{pmatrix} = 4!$$

But it is not very concentrated.

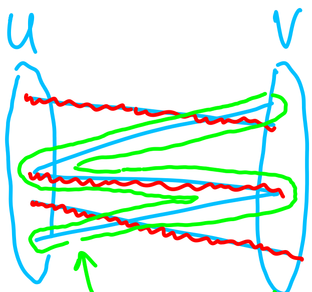
$\mathbb{E}[\det(B)^2]$

augmenting path



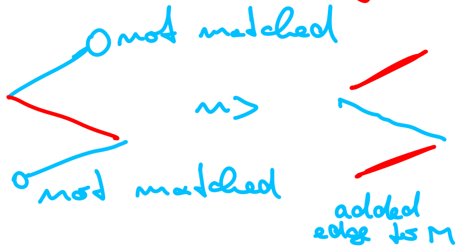
$|M| = 2$

but no short aug. path!
(we have small deg.)



augmenting path

M... matching
$|M| = 3$
? is there a larger matching?



if you know # of machings of size k

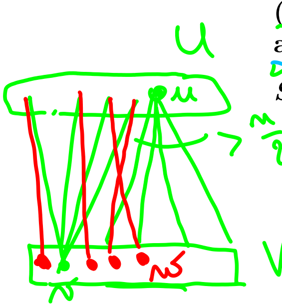
↳ $\Rightarrow$ # matchings of size $k+1$

⋮

n (perfect m.)

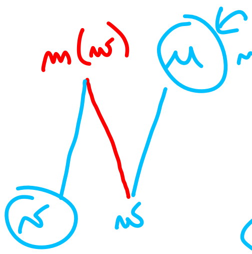
Hall theorem

6. **Let $G = (U \cup V, E)$ be a bipartite graph such that $|U| = |V| = n$ and $\boxed{\delta(G) > n/2}$ (the least degree). Show that for any matching of size at most $n-1$ there is an augmenting path of length at most 3.**



Solution:

- Consider a vertex $u \in U$ and a vertex $v \in V$ that have no matching.
- Without loss of generality u, v is not an edge (otherwise we have an augmenting path of length 1).
- Consider all neighbours of u, that is $N(u)$, again all of those have a matching otherwise we are done.
- We know that $|N(v)| > n/2$ and all of $N(u)$ have a matched vertex and also $|N(u)| > n/2$.
- Thus there is a vertex $w \in m(N(v)) \cap N(u)$ (where $m(x)$ is the vertex matched with the vertex x).
- $u, w, m(w), v$ is an augmenting path of length at most three.



tomorrow: this allows us to do a rand walk on all matchings!

↳ long enough walk ⇝ close to unif distr. ⇝ we can sample

we have a matching ⇝ using a short augm. path we get another matching

M — M', M''

THM (HALL) G has perfect matching

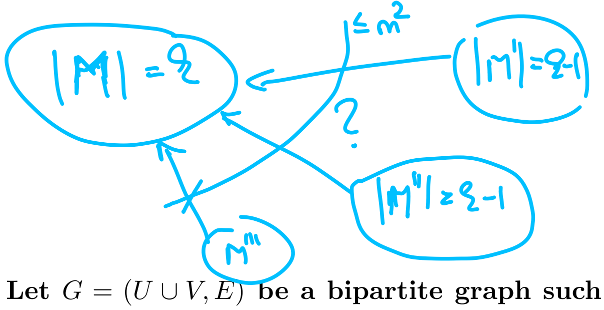
$\Leftrightarrow$

$\forall S \subseteq V \quad |N(S)| \geq |S|$

$\delta(G) > \frac{n}{2}$ $\quad |S| \leq \frac{n}{2}$ ✓

$|S| > \frac{n}{2}$ ✓ nothing on the other side misses S

using aug ≤ 3



7. **Let $G = (U \cup V, E)$ be a bipartite graph such that $|U| = |V| = n$ and $\boxed{\delta(G) > n/2}$ (the least degree). Show that for any $2 \leq k \leq n$ and a matching m of size k there are at most n^2 matchings m' of size $k-1$ such that we can get from m' to m using an augmenting path of length at most 3.**

Solution: There are at most n^2 edges. We can associate each augmenting path with its middle edge (and an augmenting path uniquely determines both m and m').