Derandomisation

May 7, 2026

Exercise 1.
Let $A_1, \dots A_n$ be a collection of subsets of $[n]$ such that $\sum_i 2^{1-|A_i|} < 1$.

1. Prove that there exists a 2-colouring of $[n]$ such that no set A_i is monochromatic.
2. Give a deterministic algorithm constructing such colouring in time polynomial in $n+m$.
3. Could we have used the method of small probability spaces?

Solution 1.

1. Let $X_1, \dots, X_m$ be independent uniform $\{0,1\}$ random variables. Let Y be the number of monochromatic set A_i. We have $\mathbb{E}[Y] = \sum_i 2^{1-|A_i|} < 1$ by linearity of expectation and mutual independence. Hence by Markov's inequality, there exists a colouring with no monochromatic set A_i.
2. We use the method of conditional expectation. For every i and j,

$$\mathbb{P}(A_i \text{ is monochromatic} | X_1 = x_1, \dots X_j = x_j) = \begin{cases} 0 & \text{if there exists } a, b \in [j] \cap A_i \text{ such that } x_a = x_b \\ 2^{1-|A_i|} & \text{if } [j] \cap A_i = \emptyset \\ 2^{-r} & \text{if all } a \in [j] \cap A_i \text{ have equal colour } x_a \\ & \text{and } |A_i \setminus [j]| = r \end{cases}$$

   This conditional probability can be computed in polynomial time, hence, $\mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j]$ can be computed in polynomial time. As

$$\begin{aligned} \mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j] &= \frac{\mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j, X_{j+1} = 1] + \mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j, X_{j+1} = 0]}{2} \\ &\geq \min\{\mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j, X_{j+1} = 1], \mathbb{E}[Y | X_1 = x_1, \dots X_j = x_j, X_{j+1} = 0]\} \end{aligned}$$

   One can choose a value of X_{j+1} that keeps the conditional expectation of Y below 1.
3. Without any additional assumption on the maximum size of the set A_i, no.

1 Deterministic $1/2$-approximations for the Maxcut problem

Recall from last exercise session that choosing a random uniform bipartition of a graph G results in a cut of expected weight $|E(G)|/2$, which gives a very simple $1/2$-approximation for the Maxcut problem. We will call it *the simple probabilistic* $1/2$*-approximation.*

Exercise 2. Conditional probabilities and greedy algorithm

1. Design a greedy deterministic algorithm for the Maxcut problem. Show that this algorithm is a $1/2$-approximation.
2. What is its running time?
3. Derandomise the simple probabilistic $1/2$-approximation using the method of conditional probabilities.

4. Compare the two algorithms.

Solution 2.

1. We construct the cut $A \sqcup B$ progressively. At the beginning, $A = B = \emptyset$. Consider an arbitrary order $v_1 < \dot{v}_n$ on the vertices. At step i, let A_i and B_i be the partial cut constructed so far. Put v_i in B_i if $|N(v_i) \cap A_i| \geq |N(v_i) \cap (B_i)|$, otherwise, put v_i in A_i.
   For each vertex v_i let E_i be the set of edges v_iv_j with $j < i$ (and hence v_j is already in one of the two parts of the cut when v_i is considered). The collection $(E_i)_{i\in[n]}$ partitions $E(G)$. The weight of the cut constructed by this algorithm is $\sum_{i\in[n]} \max(|N(v_i)\cap A_i|, |N(v_i)\cap B_i|) \geq \sum_{i\in[n]} \frac{|E_i|}{2} \geq \frac{|E(G)|}{2}$. Obviously, the weight of a cut is at most the total number of edges, so this algorithm is a 1/2-approximation.

2. This algorithm runs in time $O(|E| + |V|)$ provided that G is encoded by lists of adjacencies.

3. For each vertex v_i, let X_i be the random $\{-1, 1\}$ variable such that $X_i = 1$ if $v_i \in A$ and $X_i = -1$ if $v_i \in B$. The weight W of the cut $A \sqcup B$ is $W = \sum_{v_iv_j\in E(G)} \frac{1-X_iX_j}{2}$. We have

$$\mathbb{E}[W|X_1 = x_1, \dots, X_{i-1} = x_{i-1}, X_i = 1] =$$
$$\sum_{\substack{v_jv_k\in E(G)\\ j,k<i}} \frac{1-x_jx_k}{2} + \sum_{\substack{v_jv_k\in E(G)\\ j<i<k}} \mathbb{E}\left[\frac{1-x_jX_k}{2}\right] + \sum_{\substack{v_jv_k\in E(G)\\ i<j\leq k}} \mathbb{E}\left[\frac{1-X_jX_k}{2}\right]$$
$$+ \left(\sum_{\substack{v_jv_i\in E(G)\\ j<i}} \frac{1-x_j}{2} + \sum_{\substack{v_jv_i\in E(G)\\ i<j}} \mathbb{E}\left[\frac{1-X_j}{2}\right] \right)$$

Hence,

$$\mathbb{E}[W|X_1 = x_1, \dots, X_{i-1} = x_{i-1}, X_i = 1] - \mathbb{E}[W|X_1 = x_1, \dots, X_{i-1} = x_{i-1}, X_i = -1]$$
$$= \left(\sum_{\substack{v_jv_i\in E(G)\\ j<i}} \frac{1-x_j}{2} + \sum_{\substack{v_jv_i\in E(G)\\ i<j}} \mathbb{E}\left[\frac{1-X_j}{2}\right] \right) - \left(\sum_{\substack{v_jv_i\in E(G)\\ j<i}} \frac{1+x_j}{2} + \sum_{\substack{v_jv_i\in E(G)\\ i<j}} \mathbb{E}\left[\frac{1+X_j}{2}\right] \right)$$
$$= \sum_{\substack{v_jv_i\in E(G)\\ j<i}} \frac{1-x_j}{2} - \frac{1+x_j}{2} = \sum_{\substack{v_jv_i\in E(G)\\ j<i}} -x_j = |N(v_i) \cap B_i| - |N(v_i) \cap A_i|$$

   This difference is positive if and only if v_i has more neighbours in B_i than A_i. In other words, the choice is the same as in the greedy algorithm from the first question.

Exercise 3. Method of small probability space

1. Derandomise the simple probabilistic algorithm using the method of small probability space.

2. Argue that the obtained algortihm belongs to the class NC.

Solution 3.

1. Let $X_1, \dots X_n$ be a collection of random $\{-1, 1\}$ variable, that are pairwise independent. Then, following the notation of the the previous exercise,

$$\begin{aligned}
\mathbb{E}[W] &= \sum_{v_i v_j \in E(G)} \mathbb{E}\left[\frac{1 - X_i X_j}{2}\right] && \text{by linearity of expectation} \\
&= \sum_{v_i v_j \in E(G)} \mathbb{P}(X_i = -X_j) \\
&= \frac{|E(G)|}{2} && \text{by pairwise independence}
\end{aligned}$$

   There exists a (constructable) probability space Ω of size $O(n)$ with n pairwise independent random variable (by the theorem on small probability spaces from the lecture).

   Hence, one can evaluate in parallel the value of W on each point of Ω. By first moment method, there exists $x \in \Omega$ such that $W(x) \geq \frac{|E(G)|}{2}$. The corresponding cut is determined by $(X_i(x))_{i \in [n]}$.

2. For each $x \in \Omega$, we use one processor to compute the value of $\frac{1 - X_i X_j}{2}$ for each edge $v_i v_j \in E(G)$. Then using $O(|E(G)|$ processors, we can sum these values in time $\log(|E(G)|)$: place the values of the leaves of a binary tree of depth $\log(|E(G)|)$ and use an addition gate at every internal node of the tree. We used a $O(n|E(G)|)$ processors and the computation runs in time $O(\log(n))$.

2 Congestion minimisation

Given a directed graph G and a sequence $(s_1, t_1), \dots (s_k, t_k)$ of pairwise distinct vertices, a sequence of paths $P_1, \dots P_k$, where P_i contains s_i and t_i, has congestion C if every arc of G is contained in at most C paths. Note that $C = 1$ corresponds to the arc-disjoint path problem.

The Congestion minimisation problems asks to find a sequence of paths with minimal congestion.

★ Exercise 4.

1. Encode the congestion minimisation problem by an integer linear program, using one variable for each possible path between some s_i and t_i.

2. Relax this ILP into an LP.

3. (Optional) Argue that this LP can be solved in polynomial time, with a polynomial number of non-zero variables.

4. Show that by using fractional rounding, one obtains a $O(\frac{\log n}{\log \log n})$- approximation. To do so, use the following version of the Chernoff bound:
   Let X be a sum of independent random variables X_i, such that each $X_i \in [0, 1]$ and $\mathbb{E}[X] \leq \mu$.

$$\mathbb{P}(X \geq (1 + \alpha)\mu) \leq \exp\left(-\mu((1 + \alpha)\ln(1 + \alpha) - \alpha)\right) \quad \forall \alpha > 0$$

5. (Optional) What pessimistic estimator could we use to derandomise this algorithm using the method of conditional probabilities?

Hint : Adapt the proof of the Chernoff bound.

Notations: For every i, denote $\mathcal{P}_i$ the set of paths between s_i and t_i. For every path P between s_i and t_i, let X_P^i be the random binary variable indicating whether the path P is chosen by the randomised rounding. For every arc uv, let $Y_{uv} = \sum_i \sum_{P \ni \mathcal{P}_i} X_p^i$ the congestion of the arc uv.

★ Solution 4.

1. The idea is to consider one binary variable x^i_P for each path P in the collection $\mathcal{P}_i$ of paths from s_i to t_i. This gives the following integer linear program:

$$\begin{aligned}
\text{minimise} \quad & C & \\
\text{subject to:} \quad & \sum_{P \in \mathcal{P}_i} x^i_P = 1 & \text{for every } i \in [k] \\
& \sum_{P \ni uv} x^i_P \leq C & \text{for every } uv \in E(G) \\
& x^i_p \in \{0,1\} & \text{for every } P
\end{aligned}$$

2. We replace the last condition of the ILP by $x^i_P \in [0,1]$.

3. By the strong duality theorem, the optimal solution of this LP is equal to the optimal solution of its dual. The dual program has a polynomial number of variables and an exponential number of constraints. Thus is can be solved in polynomial time, for example by using the ellipsoid method. The solution saturates a polynomial number of constraints, so the primal solution has a polynomial number of non-zero variables. Let $(x^i_P)_{i \in [k], P \in \mathcal{P}_i}$ be a solution of the LP, with weight $C^\star$.

4. For every i, for every P, we choose one of the paths $P \in \mathcal{P}_i$ with probability x^i_P. Let X^i_P be the binary variable indicating whether the path P was chosen for (s_i, t_i). For every arc uv, let Y^i_{uv} be the number of chosen paths $P \in \mathcal{P}_i$ that use uv and let $Y_{uv} \sum_i Y^i_{uv}$. The expectation of Y_{uv} is $\mathbb{E}[Y_{uv}] = \sum_i \mathbb{E}[Y^i_{uv}] = \sum_{P \ni uv} x^i_P \leq C^\star$ by linearity of expectation.

   For every fixed arc uv, we can bound the probability that more that $\lambda C^\star$ paths use uv using the Chernoff bound. Taking $\mu = C^\star \geq \mathbb{E}[Y_{uv}]$, we have

$$\begin{aligned}
\mathbb{P}(Y_{uv} > \lambda C^\star) &\leq \exp\left(-C^\star(\lambda \ln \lambda - (\lambda - 1)\right) & \\
&\leq \exp\left(-\lambda \ln \lambda + (\lambda - 1)\right) & \text{because } C^\star \geq 1 \\
&\leq \exp\left(-6\frac{\ln n}{\ln \ln} \cdot (\ln \ln n - \ln \ln \ln n) + 6\frac{\ln n}{\ln \ln n}\right) & \text{taking } \lambda := 6\frac{\ln n}{\ln \ln n} \\
&\leq \exp\left(-\frac{6 \ln n}{2}\right) & \\
&\leq \frac{1}{n^3} &
\end{aligned}$$

   By taking union bound on all the arcs of G, the rounding results with probability at least $1 - 1/n$ in a collection of paths with congestion at most $O(\frac{\ln n}{\ln \ln n})C^\star$. In particular, the expected congestion is at most $6\frac{\ln n}{\ln \ln n}C^\star(1 - 1/n) + k/n \leq 6\frac{\ln n}{\ln \ln n}C^\star(1 - 1/n) + 1 = O(\frac{\ln n}{\ln \ln n})C^\star$.