Algorithmic Combinatorics: Lesson Plan

3.2.2015

I. Algebraic remarks

A. Polynomials $\longrightarrow$ polynomial functions

1. $p \equiv 0 \iff p(x) = 0 \ \forall x$.

B. Zeroes of univariate polynomials

~~2.~~ $\cancel{p \in \mathbb{F}[X], p \not\equiv 0 \Rightarrow |Z|}$

1. Zero-set: $Z(p) = \{x : p(x) = 0\}$.

2. If $p \in \mathbb{F}[X]$, $p \not\equiv 0$, then $|Z| \leq \deg(p)$

   a. If $\mathbb{F} = \mathbb{C} \iff$ F.T.A.

   b. For other fields, divide and induct.

II. Bipartite matchings

A. Permutations

1. Every $\pi \in S_n$ representations a possible matching:

$$\{\{u_i, v_{\pi(i)}\} : 1 \leq i \leq n\}.$$

B. Permanents & Determinants

1. Adjacency matrix: $a_{ij} = \begin{cases} 1 & \text{if } \{u_i, v_j\} \in E \\ 0 & \text{o/w} \end{cases}$

2. $\operatorname{per}(A) = \sum_{\pi} \prod_{i=1}^{n} a_{i\pi(i)}$ = # of perfect matchings in G.

   a. Hard to compute

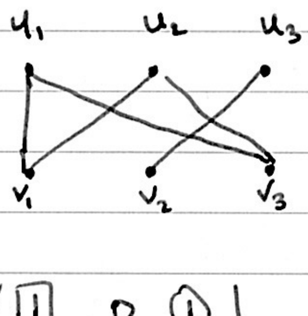
3. $\det(A) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} a_{i\pi(i)}$ = signed # p.m. in G

   a. Easy to compute

   b. Can have cancellation

4. Example:

   1. Graph



   3. per A = $\boxed{1} + 1 = 2$.

   4. det A = $1 - \boxed{1} = 0$.

   2. Matrix

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

C. Polynomial determinant

1. Variables

   a. To avoid cancellation, introduce a variable for each edge.

   b. $\tilde{a}_{ij} = \begin{cases} x_{ij} & \text{if } \{u_i, v_j\} \in E \\ 0 & \text{o/w} \end{cases}$

2. Determinant

   a. $\det(\tilde{A}) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} \tilde{a}_{i\pi(i)}$

   $$= \sum_{\substack{\pi \in \text{p.m.} \\ \text{in } G}} \operatorname{sgn}(\pi) \prod_{i=1}^{n} x_{i\pi(i)}.$$

   b. $p = \det(\tilde{A}) \in \mathbb{F}[\{x_{ij} : \{u_i, v_j\} \in E\}]$ is a polynomial

      (i) $\deg(p) = n$ if $\exists$ p.m. [else $p \equiv 0$].

      (ii) ~~only~~ "oracle access": can substitute values and compute determinant.

3. Key proposition.

   a. <u>Prop:</u> $\det(\tilde{A}) \not\equiv 0$ iff G has a perfect matching.

   b. <u>Pf:</u> (i) $\Leftarrow$: • Let σ give a perfect matching.

      • Set $x_{ij} = \begin{cases} 1 & \text{if } j = \sigma(i) \\ 0 & \text{o/w} \end{cases}$

      switch these!

      • $\det(\tilde{A}) = \sum_{\pi} \operatorname{sgn}(\pi) \prod_{i=1}^{n} \tilde{a}_{i\pi(i)}$

      $= \operatorname{sgn}(\sigma) \in \{\pm 1\}$ ✓

      $\Rightarrow \det(\tilde{A}) \not\equiv 0$.

      (ii) $\Rightarrow$: • $\det(\tilde{A}) \not\equiv 0 \Rightarrow \exists$ monomial w/ nonzero coefficient. ✓

      • monomial $\longleftrightarrow$ perfect matching ✓ $\square$

4. Randomised algorithm

   a. Need to determine if $\det(\tilde{A}) \equiv 0$

   b. By Schwartz-Zippel, picking a random point from $[2n]^m$ is correct w/ prob. $\geq \frac{1}{2}$.

c. Calculation over $\mathbb{R} \Rightarrow$ could have values as large as $(2n)^n$.

d. $\therefore$ easier to implement over $\mathbb{F} = \mathbb{F}_p$, $2n \le p < 4n$.

D. Matchings in general graphs

1. A bit more involved, but same idea
2. Details to be worked out in HW 13 Ex 6.

III. Two-colourable hypergraphs

A. Motivation

1. Saw use of probability for deterministic problems
2. Prob. method often very useful in combinatorics
3. Prob. method (existence) $\longrightarrow$ randomised algorithm (random construction) $\longrightarrow$ derandomisation (explicit construction)
4. Example: two-colourability of hypergraphs.

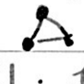
B. Two-colouring hypergraphs

1. Def^n: A k-uniform hypergraph (k-graph) is a family of sets, each with k elements.
2. Def^n: • A two-colouring of a k-graph is a red/blue colouring of its elements ~~red~~ such that each k-set has a red & blue element. (require $k \ge 2$).

~~3. Eg: $k=2$: 2-graph $\Leftrightarrow$ graph two-colourable $\Leftrightarrow$~~ • k-graph is two-colourable iff it has a 2-colouring.

3. Ex: $k=2$: 2-graph $\longleftrightarrow$ graph
two-colourable $\longleftrightarrow$ bipartite.

C. Question

1. Question: How small can a non-two-colourable k-graph be?
2. (Real-time exercise: $k=2$?)
   a. Ans: triangle: 
3. Def^n: $m(k) = \min\{|\mathcal{F}| : \mathcal{F}$ a non-two-colourable k-graph$\}$
   a. $m(2) = 3$.

D. Upper bounds.

1. Intuition

   a. Why does triangle work for $k=2$?

      (i) 3 vxs $\Rightarrow$ must have two of the same colour

      (ii) every pair an edge $\rightarrow$ monochromatic edge.

2. Generalise

   a. If we have $2k-1$ elements $\rightarrow$ ~~k~~ k have same colour

   b. $\rightsquigarrow$ monochromatic edge if all edges present

   c. $\Rightarrow m(k) \leq \binom{2k-1}{k} \sim \frac{4^k}{c\sqrt{k}}$.

3. Improvement

   a. HW $\rightarrow$ random arg. showing $m(k) \leq Ck^2 2^k$.

E. Lower bound

1. Try randomly colouring elements red / blue

   a. Intuition: very unlikely that a set should be monochromatic (when k is large).

2. <u>Thm</u> (Erdős, 1963)

   $m(k) \geq 2^{k-1}$.

   a. <u>Pf</u>: • Colour elements $\left\{ \begin{array}{l} \text{red} \\ \text{blue} \end{array} \right.$ w/ pr $\frac{1}{2}$, independently

      • Define events E_F $\forall F \in \mathcal{F}$,

      $E_F = \{F \text{ is monochromatic}\}$.

      • $\mathbb{P}(E_F) = 2 \cdot 2^{-k} = 2^{-(k-1)}$.

      • $\mathbb{P}(\text{two-colouring}) = \mathbb{P}\left(\bigcap E_F^c\right)$

      $= \mathbb{P}\left(\left(\bigcup E_F\right)^c\right) = 1 - \mathbb{P}\left(\bigcup E_F\right)$

      $\geq 1 - \sum_F \mathbb{P}(E_F) = 1 - |\mathcal{F}| \cdot 2^{-(k-1)} > 0$

      if $|\mathcal{F}| < 2^{k-1}$

      • $\Rightarrow \exists$ proper two-colouring. $\Rightarrow m(k) \geq 2^{k-1}$. $\square$

3. Derandomisation

   a. Proof shows <u>existence</u> of two-colouring

   b. We would like to know <u>how</u> to two-colour it.

F. Maker-Breaker

1. Two players
   a. We imagine there are two people: M & B
      (i) M colours elements red (M goes first)
      (ii) B colours elements blue
   b. They go turn-by-turn, colouring one element each time
      (i) No reason to favour one colour over the other
2. Objectives
   a. If both M & B manage to claim an element from each $F \in \mathcal{F}$ $\longrightarrow$ proper 2-colouring
   b. Now suppose they were antagonistic instead of cooperative
      (i) B still tries to claim <u>one</u> element from <u>each</u> $F \in \mathcal{F}$.
      (ii) M tries to stop B from succeeding; i.e. tries to claim <u>all</u> elements from <u>one</u> $F \in \mathcal{F}$.
      (iii) Whoever succeeds $\rightarrow$ winner.
   c Example of a "positional game"
      (i) Other examples: Tic-Tac-Toe, Hex.
3. Maker-Breaker $\Longrightarrow$ Two-colouring
   a. Suppose we could guarantee that if $|\mathcal{F}| \le m$, then B wins.
      (i) $\Rightarrow$ B has a strategy s.t. no matter what M does, B can choose elements s.t. she always claims an element from every $F \in \mathcal{F}$.
   b. ~~E~~ Then M uses B's winning strategy too
      (i) Makes arbitrary first move
      (ii) B's strategy $\Rightarrow$ <u>both</u> players claim an elt from each $F \in \mathcal{F}$.
      (iii) $\Rightarrow$ each $F \in \mathcal{F}$ has red & blue elt $\Rightarrow$ two-coloured
      (iv) $\Rightarrow m(k) > m$.

G. Erdős – Selfridge

1. Thm: (Erdős–Selfridge, 1973)

   If $|\mathcal{F}| < 2^{k-1}$, B ~~[illegible]~~ has a winning strategy.

   a. Remarks

      (i) $\Rightarrow m(k) \geq 2^{k-1}$, as before

      [but gives deterministic algorithm]

      (ii) Stronger statement, since B is playing against an antagonistic opponent.

2. Corollary of non-uniform version.

   a. Thm: Let $\mathcal{F}$ be a collection of sets. If

   $\sum_{F \in \mathcal{F}} 2^{-|F|} < \frac{1}{2}$, then B has a winning strategy.