7. Algebraic Branching Programs

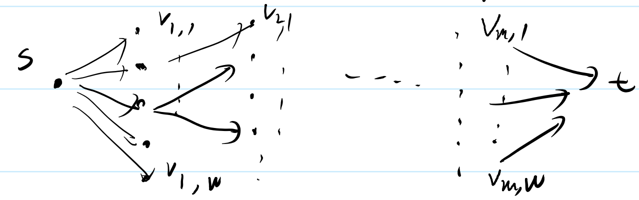
Tuesday, September 12, 2023 6:05 PM

We want to show that the polynomial $DET(X_{11}, \dots, X_{nn}) := \sum_{\sigma \in S_n} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} X_{i\sigma(i)}$ is in VP.

In fact, we will show that it is in a subclass $VBP \subseteq VP$, and is VBP-complete (to be made rigorous).

<u>Def</u>. An algebraic branching program (ABP) in variables $X_1, \dots, X_n$ over $\mathbb{F}$ is a layered (labeled) directed graph with two special vertices s and t, and layers of vertices $v_{1,1}, \dots, v_{n,w}$.

(Analogue of Boolean branching programs)



w = width of the VBP

$+1$ = length of paths from s to t
= length (or depth) of the VBP

size = # vertices + # edges

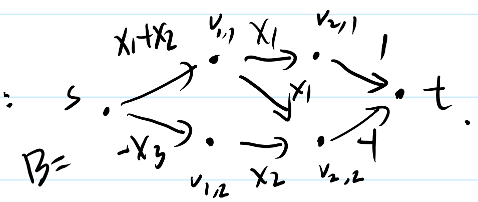
edges are labeled with polynomials of degree ≤ 1 in $X_1, \dots, X_n$, i.e. $\sum c_i X_i + c_0$

$w(e)$ = label of e.

The polynomial the ABP computes is

$$\sum_{\text{path } p: s \to t} \prod_{e \in p} w(e)$$

<u>Example</u>: $B =$



The polynomial that B computes can be found via dynamic programming:

$s = 1$, $v_{1,1} = s \cdot (X_1 + X_2) = X_1 + X_2$, $v_{1,2} = s \cdot (-X_3) = -X_3$.

$v_{2,1} = v_{1,1} \cdot X_1 = X_1^2 + X_1 X_2$. $v_{2,2} = v_{1,1} \cdot X_1 + v_{1,2} \cdot X_2$
$= X_1^2 + X_1 X_2 - X_2 X_3$

$t = v_{2,1} \cdot 1 + v_{2,2} \cdot (-1) = X_2 \cdot X_3$

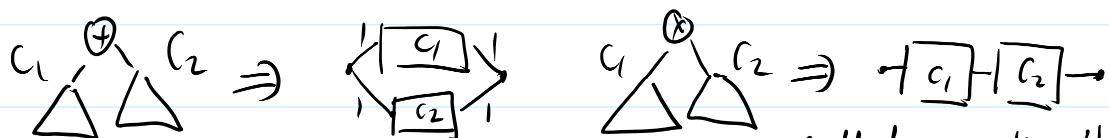
width = 2, length = 3

<u>Def</u> VBP = set of polynomial families (f_n) in $X_1, \dots, X_n$ computed by poly(n)-size ABPs. (Note: $\deg(f_n) \le \mathrm{poly}(n)$)

Claim: $VF \subseteq VBP \subseteq VP$.

<u>Pf</u>: It is easy to see that a poly(n)-size ABP can be simulated by a poly(n)-size circuit. So $VBP \subseteq VP$.

Also note that a poly(n)-size formula can be simulated by a poly(n)-size ABP:



(Add dummy vertices if necessary).

So $VF \subseteq VBP$

(Add dummy vertices if necessary).

So $VF \subseteq VBP$ $\square$

Completeness.

(Valiant)

Def: 1. A function $t: \mathbb{N} \to \mathbb{N}$ is p-bounded if $t(n) \le n^c$ for some $c > 0$.

2. $f(x_1, \dots, x_n)$ is a projection of $g(Y_1, \dots, Y_m)$ if $f = g(\pi(Y_1), \dots, \pi(Y_m))$ where $\pi(Y_i) \in \{x_1, \dots, x_n\} \cup \mathbb{F}$ for $i = 1, \dots, m$. Denote this by $f \le_p g$

3. For polynomial families (f_n) and (g_n), we say (f_n) is a p-projection of (g_n) if $f_n \le_p g_{t(n)}$ for some p-bounded function t and all n.
Denote this by $(f_n) \le_p (g_n)$

Note: 1. for many classes C, including VF, VBP, and VP, we have $\begin{matrix}(g_n) \in C \ \& \\ (f_n) \le_p (g_n)\end{matrix} \Rightarrow (f_n) \in C$.

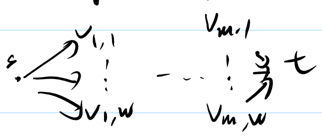
2. $(f_n) \le_p (g_n)$, $(g_n) \le_p (h_n) \Rightarrow (f_n) \le_p (h_n)$.

Def: (f_n) is said to be C-complete if $(f_n) \in C$ and $(g_n) \le_p (f_n)$ for all $(g_n) \in C$.

Example (Iterated Matrix Multiplication).

$IMM_{w,\ell}$ = the $(1,1)$-entry of $\prod_{i=1}^{\ell} M^{(i)}$, where $M^{(i)} = (x_{j,k}^{(i)})_{1 \le j,k \le w}$ in the variables $x_{j,k}^{(i)}$

Claim: $IMM_{n,n}$ is VBP-complete.

Proof: For an ABP , let $u = (w(s, v_{1,1}), \dots, w(s, v_{1,w}))$

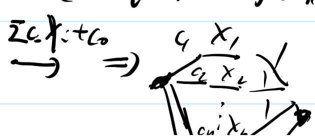
$$v = \begin{pmatrix} w(v_{m,1}, t) \\ \vdots \\ w(v_{m,w}, t) \end{pmatrix}$$

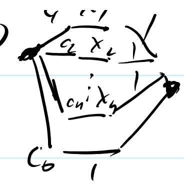
$$M^{(i)} = (w(v_{i,j}, v_{i+1,k}))_{1 \le j,k \le w}.$$

Then the ABP computes $u M^{(1)} \cdots M^{(m-1)} v$.

Using this characterization, can turn IMM into an ABP.

Conversely, given an ABP we may assume all the weights are in $\{x_1, \dots, x_n\} \cup \mathbb{F}$.

(By increasing the size of the ABP if necessary: 

(By increasing the size of the ABP if necessary: $\longrightarrow \Longrightarrow$ )

Then the ABP computes a projection of $IMM_{m,m}$, $m \le poly(n)$. $\square$

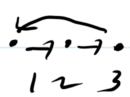
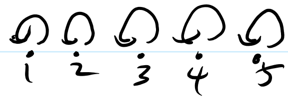
<u>Thm</u> (Mahajan-Vinay '97) DET is VBP-complete.

We first show $(f_n) \le (DET_n)$ for all $(f_n) \in VBP$.

(Note: # variables in DET is a square. May assume $DET_n = 0$ if n is not a square)

Suppose G is a directed graph on $\{1, \dots, n\}$ with weight function w on its edges such that each vertex i has a self-loop with weight 1.

Each permutation $\sigma \in S_n$ corresponds to a cycle cover in the complete graph with n self-loops, i.e. a disjoint union of simple cycles covering all the vertices $\{1, \dots, n\}$.

E.g. $(123)(45) \Longleftrightarrow$   identity $\Longleftrightarrow$ 

Let $A_G = (w(i,j))_{1 \le i,j \le n}$.

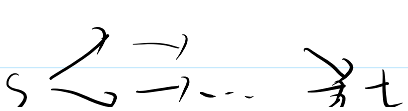
<u>Lemma</u>: $$\det(A_G) = \sum_{\text{cycle cover } C} sgn(C) \prod_{e \in C} w(e)$$

where $sgn(C) = (-1)^{\#\text{even cycles in } C}$. (an even cycle is a cycle whose # vertices (or edges) is even).

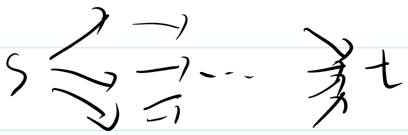
Pf: This holds by definition.

<u>Note</u>: Similarly, $perm(A_G) = \sum_{\text{cycle cover } C} \underbrace{\prod_{e \in C} w(e)}_{w(C)}$.

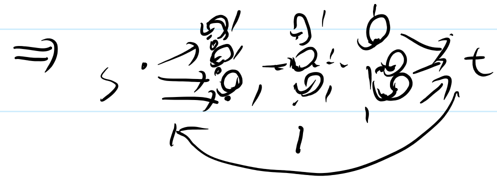
Given an ABP G, modify it into another graph G' as follows:



1. add an edge $t \to s$ with weight 1.
2. add a self-loop $v \to v$ with weight 1 for $v \in G$



1. add an edge $t \to s$ with weight 1.
2. add a self-loop $v \to v$ with weight 1 for $v \in G$, $v \notin \{s, t\}$.



Then cycle cover C in G' $\Longleftrightarrow$ path $p: s \rightsquigarrow t$ in G

$$w(C) = w(p)$$

$\mathrm{sgn}(C)$ is the same for all cycle covers C, depending only on the parity of the length of the ABP G.

So G computes $\displaystyle\sum_{\substack{\text{path } p: s \rightsquigarrow t \\ \text{in } G}} w(p) = \pm \sum_{\substack{\text{cycle cover } C \\ \text{of } G'}} w(C) = \pm \det(A_{G'})$.

So every $(f_n) \in \mathrm{ABP}$ is a p-projection of (DET_n) $\square$

<u>Remark</u>: In the above, we also have $\mathrm{perm}(A_{G'}) = \displaystyle\sum_{p: s \rightsquigarrow t} w(p)$.
It differs from $\det(A_{G'})$ by ± 1 in this case.

We still need to show $\mathrm{DET} \in \mathrm{VBP}$.

Idea: Use dynamic programming to compute $\mathrm{DET} = \displaystyle\sum_{\substack{\text{cycle cover } C \\ \text{of } G'}} \mathrm{sgn}(C)\, w(C)$.

However, the simple cycle property is not compatible with dynamic programming.

Idea: replace simple cycles by closed walks ("clows").

Will do this next time.