9. Permanent is VNP-Complete, Part 1

Sunday, September 17, 2023 8:26 PM

$\mathrm{PERM}_n = \mathrm{perm}((X_{ij})) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} X_{i\sigma(i)} \in \mathbb{F}[X_{11}, \ldots, X_{nn}]$.

$$\mathrm{VNP} = \left\{ (f_n) : \forall n,\ f_n = \sum_{(e_1,\ldots,e_{t(n)}) \in \{0,1\}^{t(n)}} g_{n+t(n)}(X_1,\ldots,X_n,e_1,\ldots,e_{t(n)}),\ t: \mathbb{N}\to\mathbb{N} \text{ is p-bounded}, (g_n) \in \mathrm{VP} \right\}.$$

<u>Thm 1</u> (Valiant): PERM is VNP-complete (under p-projections)

<u>Prop 1.</u> PERM $\in$ VNP.

<u>Proof 1</u>: (Ryser's formula): For an $n\times n$ matrix $A = (a_{ij})$,

depth 3, unbounded formula

$$\mathrm{perm}(A) = (-1)^n \sum_{S \subseteq \{1,\ldots,n\}} (-1)^{|S|} \prod_{i=1}^{n} \sum_{j\in S} a_{ij}.$$

Proof uses the inclusion-exclusion principle. (Ref: Lint-Wilson)

$$\text{So } \mathrm{PERM}_n = (-1)^n \sum_{(e_1,\ldots,e_n)\in\{0,1\}^n} \left(\prod_{i=1}^{n}(1-2e_i)\right)\left(\prod_{i=1}^{n} \sum_{j=1}^{n} X_{ij}\cdot e_j\right)$$

$$\text{And } (-1)^n\left(\prod_{i=1}^{n}(1-2Y_i)\right)\cdot\left(\prod_{i=1}^{n}\sum_{j=1}^{n} X_{ij} Y_j\right) \in \mathrm{VP}. \qquad \square$$

<u>Proof 2</u>: Let $E(Y_{11},\ldots,Y_{nn})$ be a poly-size algebraic formula such that for $(e_{11},\ldots,e_{nn}) \in \{0,1\}^{n^2}$, $E(e_{11},\ldots,e_{nn}) = 1$ iff (e_{ij}) is a permutation matrix.

This can be done by writing down a Boolean formula and converting it into an algebraic formula.

$$\text{Then } \sum_{(e_{11},\ldots,e_{nn})\in\{0,1\}^{n^2}} E(e_{11},\ldots,e_{nn}) \cdot \prod_{1\le i,j\le n} (\underbrace{X_{ij} e_{ij} + 1 - e_{ij}}_{= \begin{cases} 1 & \text{if } e_{ij}=0 \\ X_{ij} & \text{if } e_{ij}=1 \end{cases}})$$

$$= \sum_{\sigma\in S_n}\prod_{i=1}^{n} X_{i\sigma(i)}.$$

$$\text{And } E(Y_{11},\ldots,Y_{nn}) \prod_{1\le i,j\le n} (X_{ij}Y_{ij} + 1 - Y_{ij}) \in \mathrm{VP}. \qquad \square$$

It remains to prove:

<u>Prop 2</u>: $(f_n) \le_p (\mathrm{PERM}_n)$ for every $(f_n) \in \mathrm{VNP}$.

Define $\mathrm{VNP}_0 := \{ (f_n) : \forall n,\ f_n = \sum g_{n+t(n)}(X_1,\ldots,X_n,e_1,\ldots,e_{t(n)}),\ t:\mathbb{N}\to\mathbb{N} \text{ is p-bounded} \}$

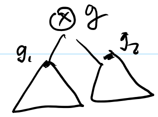
Define $VNP_e := \left\{ (f_n) : \forall n,\ f_n = \sum_{(e_1,\dots,e_{t(n)}) \in \{0,1\}^{t(n)}} g_{n+t(n)}(X_1,\dots,X_n, e_1,\dots,e_{t(n)}),\ t: \mathbb{N} \to \mathbb{N} \text{ is } p\text{-bounded},\ (g_n) \in VF \right\}$.

(VF is also called VP_e)

Thm 2 : $VNP = VNP_e$

Our proof of Thm 2 follows (Malod – Portier '08), which is also in "determinant versus permanent" by Bläser.

Def: A circuit is multiplicatively disjoint (MD) if for every multiplication gate g, the two subcircuits of the two children of g are disjoint: i.e., $\nexists h$ with paths $h \rightsquigarrow g_1$ and $h \rightsquigarrow g_2$.



MD circuits interpolate circuits and formulas.

Lemma 1: Suppose f is computed by a homogeneous circuit C of size s, and $\deg(f) = d$. Then f is computed by a MD circuit C' of size $\le s\cdot(d+1)$.

Pf: For technical reasons, assume C does not have constant input gates. Instead, we allow $\oplus$ with children h_1, h_2 to compute $c_1 \cdot h_1 + c_2 \cdot h_2$, and $\otimes$ with children h_1, h_2 to compute $c \cdot h_1 \cdot h_2$, $c, c_1, c_2 \in \mathbb{F}$.

For each $g \in C$ of degree e, we construct $g_1, \dots, g_{d-e+1} \in C'$, where i is called the index of g_i, such that for $i = 1, \dots, d-e+1$:

(1) the gate g_i computes the same polynomial as g.

(2) Suppose $h_j \in C'$ is in the subcircuit of g_i. Then the index j of h_j satisfies $i \le j \le i+e-1$.

(3) the subcircuit of g_i is MD.

Construct g_i's in an inductive, bottom-up fashion:

1. Suppose g is an input gate. Then $e = \deg(g) = 1$.
   In this case, just let $g_1, \dots, g_{d-e+1}$ be copies of g.

2. Now suppose $g = c_1 \cdot u + c_2 \cdot v$. Then $\deg(u) = \deg(v) = e$.

   Just let $g_i = c_1 u_i + c_2 v_i$ for $i = 1, \dots, d-e+1$.
   Then (1) – (3) hold for g_i by the induction hypothesis.

Then (1) – (3) hold for g_i by the induction hypothesis.

3. Now suppose $g = c \cdot u \cdot v$, $\deg(u) = e_1$, $\deg(v) = e_2$. Then $e = e_1 + e_2$.

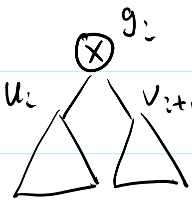
For $i = 1, \dots, d-e+1$, let $g_i := c \cdot u_i \cdot v_{i+e_1}$

u_i is a copy of u

To see this is well-defined, note $1 \le i \le d-e+1 \le d - e_1 + 1$

v_{i+e_1} is a copy of v.

$1 \le i+e_1 \le (d-e+1)+e_1 = d - e_2 + 1$



(1): By the induction hypothesis, g_i computes the same polynomial as g.

(2): If h_j is in the subcircuit of u_i, then $i \le j \le i+e_1-1 \le i+e-1$.

If h_j is in the subcircuit of v_{i+e_1}, then $i+e_1 \le j \le i+e_1+e_2-1 = i+e-1$.

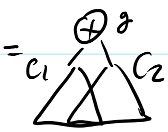
So h_j is in the subcircuit of $g_i \Rightarrow i \le j \le i+e-1$.

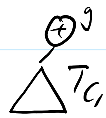
(3): Assume the subcircuit of g_i is not MD. As both the subcircuits of u_i and that of v_{i+e_1} is MD, we must have a gate h_j in both the subcircuit of u_i and that of v_{i+e_1}

Then $\left.\begin{array}{l} i \le j \le i+e_1-1 \\ \text{and } i+e_1 \le j \le i+e-1 \end{array}\right\}$ contradiction. So the subcircuit of g_i is MD.

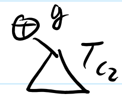
Continue this process. We get a MD circuit computing f of size $\le s \cdot (d+1)$ $\square$

Def: A *parse tree* of a circuit C is recursively defined as follows:

- If C is a node, it is itself the only parse tree of C.
- If $C =$ 

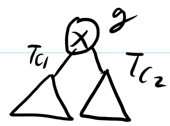
, a parse tree of C is 

T_{C_1} is a parse tree of C_1

or 

T_{C_2} is a parse tree of C_2.

- If $C =$ 

, a parse tree of C is 

.

Define $pt(C)$ to be the set of parse trees of C.

For a parse tree T, $w(T)$ is the product of the labels of the leaves of T (with multiplicities)

Claim: $C = \sum_{T \in pt(C)} w(T)$ as a polynomial.

Claim: $C = \sum_{T \in pt(C)} w(T)$ as a polynomial.

Pf: We prove the claim by induction. When C is a node, the claim is obvious.

If $C = C_1 + C_2$, then $C = C_1 + C_2 \overset{\text{ind. hypotheses}}{=} \sum_{T \in pt(C_1)} w(T) + \sum_{T \in pt(C_2)} w(T)$

$$\overset{\text{def.}}{=} \sum_{T \in pt(C)} w(T).$$

If $C = C_1 \times C_2$, then $C = C_1 \times C_2 \overset{\text{ind. hypotheses}}{=} \left(\sum_{T \in pt(C_1)} w(T) \right) \cdot \left(\sum_{T \in pt(C_2)} w(T) \right)$

$$= \sum_{(T_1, T_2) \in pt(C_1) \times pt(C_2)} w(T_1) \cdot w(T_2)$$

$$\overset{\text{def}}{=} \sum_{T \in pt(C)} w(T). \qquad \square$$

Observation: Let C be a circuit such that every gate has a path to the output gate. Then C is MD iff every $T \in pt(C)$ is a subcircuit of C.

Lemma 2: Let $C \overset{=}{} C(x_1, \ldots, x_n)$ be a MD circuit s.t. every gate has a path to the output gate. Let E be the edge set of C. For each $e \in E$, let x_e be a variable. There exists a formula F in $x_1, \ldots, x_n$ and $\{x_e\}_{e \in E}$ of size polynomial in size(C) such that for $a = (a_e) \in \{0,1\}^E$,

$$F(x_1, \ldots, x_n, a) = \begin{cases} w(T) & \text{if the edges selected by } a \text{ form a parse tree } T \text{ of } C. \\ 0 & \text{otherwise.} \end{cases}$$

Pf: We say a gate g is <u>selected</u> by $a \in \{0,1\}^E$ if either g is the output gate, or $(g, \text{parent of } g)$ is selected by a.

One can build a poly-size Boolean formula $F_0 : \{0,1\}^E \to \{0,1\}$ that evaluates to 1 at a iff the following condition holds:

- for every gate g selected by a, if g is an addition gate, for every gate g selected by a, if g is an addition gate, then (h,g) is selected by a for exactly one child h of g, and if g is a multiplication gate, then (h,g) is selected by a for all children h of g.

Turn F_0 into an algebraic circuit F_1 in $\{X_e\}_{e\in E}$ such that F_1 agrees with F_0 on $\{0,1\}^E$.

$$\text{Let } F = F_1 \cdot \prod_{\substack{e=(u,v)\\ u \text{ is an input gate}}} \underbrace{\left(X_{(u,v)}\cdot \overset{\text{label of } u}{w(u)} + 1 - X_{(u,v)}\right)}_{=\begin{cases} w(u) & \text{if } X_{(u,v)}=1 \\ 1 & \text{if } X_{(u,v)}=0\end{cases}}$$

Then F is as desired (except when C is a node, in which case we just let $F=C$). $\square$

<u>Cor</u> $VP \subseteq VNP_e$

Pf: Let $f\in VP$. By homogenization and Lemma 1, f is computed by a poly-size MD circuit C. Build the formula F in Lemma 2 for C. Then

$$C = \sum_{T\in Pt(C)} wt(T) = \sum_{a\in\{0,1\}^{E \leftarrow \text{edge set of } C}} F(x_1,\ldots,x_n,a).\quad \text{So } f\in VNP_e$$

$\square$

Proof of Thm 2 ($VNP = VNP_e$): Let $f = f(x_1,\ldots,x_n)\in VNP$.

Then $f(x_1,\ldots,x_n) = \sum_{e\in\{0,1\}^{t(n)}} g(x_1,\ldots,x_n,e_1,\ldots,e_{t(n)})$ where t is p-bounded and $g\in VP$.

As $g\in VP\subseteq VNP_e$, we may write $g(x_1,\ldots,x_n,y_1,\ldots,y_{t(n)}) = \sum_{e'\in\{0,1\}^{t'(n)}} h(x_1,\ldots,x_n,y_1,\ldots,y_{t(n)},e')$, where t' is p-bounded.

Then $f = \sum_{(e,e')\in\{0,1\}^{t(n)+t'(n)}} g(x_1,\ldots,x_n,e,e')$. So $f\in VNP_e$. $\square$

We will finish the proof of the VNP-completeness of the permanent next time.