3. Baur–Strassen Theorem

Tuesday, August 29, 2023 12:17 PM

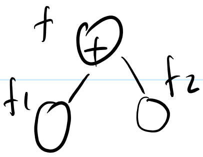
Theorem (Baur-Strassen '83): Suppose $f(x_1,\dots,x_n) \in \mathbb{F}[x_1,\dots,x_n]$ is computed by a circuit C of size s. Then $\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}$ are computed by a multi-output circuit of size $O(s)$.

Remark: Note that if $f_1,\dots,f_m \in \mathbb{F}[x_1,\dots,x_n]$ are polynomials computed by a multi-output circuit of size s, then $\sum_{i=1}^m f_i \cdot t_i \in \mathbb{F}[x_1,\dots,x_n, t_1,\dots,t_m]$ is computed by a circuit of size $s+O(t)$.

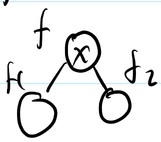
Baur-Strassen implies that we can go back: If $\sum_{i=1}^m f_i t_i$ is computed by a circuit of size s, then $f_1,\dots,f_m$ are computed by a multi-output circuit of size $O(s)$.

In particular, multi-output-ness does not help in proving circuit lower bounds: If $f_1,\dots,f_m$ require a large circuit to compute simultaneously, then $\sum_{i=1}^m f_i t_i$ also requires a large circuit (in a fine-grained sense)

Remark 2: A circuit of size s computing f can be turned into a circuit of size $O(s)$ computing $\frac{\partial f}{\partial x_i}$, for each i, via recursion:



$$\frac{\partial f}{\partial x_i} = \frac{\partial f_1}{\partial x_1} + \frac{\partial f_2}{\partial x_2}.$$



$$\frac{\partial f}{\partial x_i} = \frac{\partial f_1}{\partial x_i}\cdot f_2 + \frac{\partial f_2}{\partial x_i}\cdot f_1.$$

The nontrivial part of Baur-Strassen is saving the factor of n.

<u>Lemma</u> (chain rule in multivariate derivatives)

Let $g \in \mathbb{F}[y_1,\dots,y_m]$, $h_1,\dots,h_m \in \mathbb{F}[x_1,\dots,x_n]$. Let $f = g(h_1,\dots,h_m)$

Then $$\frac{\partial f}{\partial x_i} = \sum_{j=1}^m \frac{\partial g}{\partial y_j}(h_1,\dots,h_m)\cdot\frac{\partial h_j}{\partial x_i} \quad \text{for } i=1,\dots,n.$$

Proof sketch: At a point $a=(a_1,\dots,a_n)$, $h_j = h_j(a) + \sum_{i=1}^n \frac{\partial h_j}{\partial x_i}(a)(x_i-a_i) + \tilde{h}_j$ (1)

where $\tilde{h}_j \in \langle x_1-a_1,\dots,x_n-a_n\rangle^2$

where $\tilde{h}_j \in \langle X_1-a_1, \cdots, X_n-a_n\rangle^2$
for $j=1,\cdots,m$.

Let $b_j = h_j(a)$ and $b=(b_1,\cdots,b_m)$ for $j=1,\cdots,m$. Then $g = g(b) + \sum_{j=1}^{m} \frac{\partial g}{\partial Y_j}(b)(Y_j - b_j) + \tilde{g}$ (2)

where $\tilde{g} \in \langle Y_1-b_1,\cdots,Y_m-b_m\rangle^2$

By (2), $f = g(h_1,\cdots,h_m) = \underset{f(a)}{g(b)} + \sum_{j=1}^{m}\frac{\partial g}{\partial Y_j}(b)(h_j - h_j(a)) + \tilde{f}$ (Note $b_j = h_j(a)$)

where $\tilde{f} = \tilde{g}(h_1,\cdots,h_m) \in \langle h_1 - h_1(a),\cdots, h_m - h_m(a)\rangle^2$
$\subseteq \langle X_1-a_1,\cdots,X_n-a_n\rangle^2$.

By (1), we further have

$$f = f(a) + \sum_{j=1}^{m}\frac{\partial g}{\partial Y_j}(b)\left(\sum_{i=1}^{n}\frac{\partial h_j}{\partial X_i}(a)(X_i-a_i) + \tilde{h}_j\right) + \tilde{f}$$

$$\equiv f(a) + \sum_{i=1}^{n}\sum_{j=1}^{m}\frac{\partial g}{\partial Y_j}(b)\frac{\partial h_j}{\partial X_i}(a)(X_i-a_i) \mod \langle X_1-a_1,\cdots,X_n-a_n\rangle^2 \quad (3)$$

But we also have $f \equiv f(a) + \sum_{i=1}^{n}\frac{\partial f}{\partial X_i}(a)(X_i-a_i) \mod \langle X_1-a_1,\cdots,X_n-a_n\rangle^2$ (4)

$X_1-a_1,\cdots,X_n-a_n$ are a basis of $\{P\in \mathbb{F}[X_1,\cdots,X_n] : P(a)=0\}/\langle X_1-a_1,\cdots,X_n-a_n\rangle^2$

So $\frac{\partial f}{\partial X_i}(a) = \sum_{j=1}^{m}\frac{\partial g}{\partial Y_j}(b)\frac{\partial h_j}{\partial X_i}(a)$ for $i=1,\cdots,n$.

The same holds when $a_1,\cdots,a_n$ are replaced by variables $X_1,\cdots,X_n$.

$\Rightarrow \frac{\partial f}{\partial X_i} = \sum_{j=1}^{m}\frac{\partial g}{\partial Y_j}(h_1,\cdots,h_m)\frac{\partial h_j}{\partial X_i}$ for $i=1,\cdots,n$. $\square$

Proof of Baur-Strassen: We show size needed to compute $\frac{\partial f}{\partial X_1},\cdots,\frac{\partial f}{\partial X_n}$ is $O(\#\text{wires} + \#\text{gates})$

Induct on the size := #wires + #gates of the circuit C computing f.

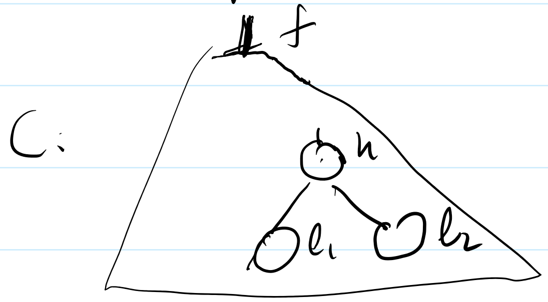
Base case: C has no non-leaf. Then f is either a constant or a variable X_{i_0}

If $f=c$, $\frac{\partial f}{\partial X_i}=0$. If $f = X_{i_0}$, $\frac{\partial f}{\partial X_i} = \begin{cases} 1 & i_0 = i \\ 0 & i_0 \neq i. \end{cases}$

Either case, $\frac{\partial f}{\partial X_1},\cdots,\frac{\partial f}{\partial X_n}$ can be computed by a circuit of size $O(n) = O(\#\text{wires} + \#\text{gates})$

... $\partial x_1, \cdots, x_n$ ... computed by a circuit of size $O(n) = O(\#\text{wires} + \#\text{gates})$

Induction Step: Suppose C has a nonleaf h, We may assume $h = l_1 + l_2$ or $l_1 \times l_2$ where l_1, l_2 are leaves i.e. constants or variables.



Let C' be the circuit obtained from C by replacing h by a new input variable Y.
where the wires between h and l_1, and h and l_2 are removed.

So $\text{size}(C') \leq \text{size}(C) - 1$.

Let $g \in \mathbb{F}[X_1, \cdots, X_n, Y]$ be the polynomial that C' computes

Then $f = g(x_1, \cdots, x_n, h)$.

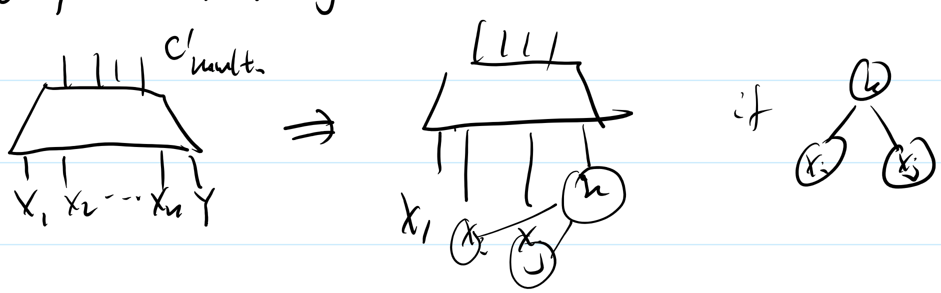
So for $i = 1, \cdots, n$,
$$\frac{\partial f}{\partial X_i} = \left(\sum_{j=1}^{n} \frac{\partial g}{\partial X_j}(x_1, \cdots, x_n, h) \cdot \delta_{ij}\right) + \frac{\partial g}{\partial Y}(x_1, \cdots, x_n, h)\, \frac{\partial h}{\partial X_i}$$
$$= \frac{\partial g}{\partial X_i}(X_1, \cdots, X_n, h) + \frac{\partial g}{\partial Y}(X_1, \cdots, X_n, h) \cdot t_i \qquad (*)$$

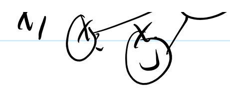
where $t_i = \frac{\partial h}{\partial X_i} \in \{0, 1, X_j\}$
$t_i \neq 0$ for constantly many i.

By the induction hypothesis, $\frac{\partial g}{\partial X_1}(X_1, \cdots, X_n, Y), \cdots, \frac{\partial g}{\partial X_n}(X_1, \cdots, X_n, Y), \frac{\partial g}{\partial Y}(X_1, \cdots, X_n, Y)$

are computed by a circuit C'_{multi} of size $O(\text{size}(C'))$

Replacing Y in C'_{multi} by h





Then we get a circuit computing $\frac{\partial g}{\partial x_1}(x_1,\dots,x_n,h), \dots, \frac{\partial g}{\partial x_n}(x_1,\dots,x_n,h), \frac{\partial g}{\partial y}(x_1,\dots,x_n,h)$.

Compute $\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}$ using $(*)$

As $t_i \neq 0$ for constantly many (≤ 2) i, the size increases by $\leq O(1)$.

$\square$.

Application: Shortest cycle

Given a directed graph G on n nodes with edge weights in $\{1, \dots, M\}$, how fast can we find a cycle with the minimum total weight?

Dijkstra for each node as the start node: $O(n^3 \log M)$

No "truly subcubic" algorithm known, deterministic or randomized.

$\rightarrow O(n^{3-c} \cdot \mathrm{polylog} M)$, $c > 0$.

Using Baur-Strassen, we will see a randomized algorithm with running time $\tilde{O}(n^{\omega} \cdot M)$, where ω is the matrix multiplication exponent ($\omega < 2.38$).

Cygan - Gabow - Sankowski, JACM '15.