ADFOCS 2017

intro & simulations

Amir Yehudayoff (Technion)

we have limited resources

time

space

energy

...

computational complexity studies the achievable

algebraic complexity theory

goal: to compute polynomials

tools: algebraic devices (use $+, \times, \div$)

field $\mathbb{F}$

variables $X = \{x_1, \ldots, x_n\}$

polynomial $f \in \mathbb{F}[X]$

what is the complexity of f?

examples

polynomials

matrix product

$$(XY)_{i,j} = \sum_{k} x_{i,k} y_{k,j}$$

determinant (linear algebra)

$$det(X) = det_n(X) = \sum_{\pi \in S_n} sign(\pi) \prod_{i \in [n]} x_{i,\pi(i)}$$

permanent (combinatorics & complexity)

$$perm(X) = perm_n(X) = \sum_{\pi \in S_n} \prod_{i \in [n]} x_{i,\pi(i)}$$

determinant

det_n has $n!$ monomials

can we compute it efficiently?

determinant

det_n has $n!$ monomials

can we compute it efficiently?

[...,Gauss,...]

(i) in $O(n^3)$ steps write $X = SU$

where $det(S) = \pm 1$ and U upper triangular

(ii) $det(X) = det(U) = \prod_i U_{i,i}$

(iii) can avoid divisions [Strassen]

permanent

$perm_n$ has $n!$ monomials

can we compute it efficiently?

permanent

$perm_n$ has $n!$ monomials

can we compute it efficiently?

[Ryser]

$$perm_n(X) = (-1)^n \sum_{T \subseteq [n]} (-1)^{|T|} \prod_{i \in [n]} \sum_{j \in T} x_{i,j}$$

(best known)

devices

circuits

straight line programs: $f_1, f_2, \ldots, f_s$

$$f_i \in X \cup \mathbb{F} \quad \text{or} \quad f_i = f_j \star f_k$$

with $j, k < i$ and $\star \in \{+, \times, \div\}$

f_s is the output

circuits: DAGs representing the SLP

costs:

size - number of gates

depth - length of longest path

ABPs

algebraic branching programs: DAG from a to b

edge e is labelled $\lambda_e \in X \cup \mathbb{F}$

$$f = \sum_{\gamma : a \to b} \prod_{e \in \gamma} \lambda_e$$

iterated matrix product:
$w \times w$ matrices $M_1, \ldots, M_\ell$ where $(M_i)_{j,k} \in X \cup \mathbb{F}$

$$f = (M_1 M_2 \ldots M_\ell)_{1,1}$$

costs:

size

length - ℓ

width - w

formulas

formulas are circuits where graph is a tree

allowed to use “subcomputations” once

costs:

size

depth

claim $\log(\mathit{size}) \leq \mathit{depth} \leq O(\log(\mathit{size}))$

complexity

devices have **costs**

polynomials have **complexities**

e.g. circuit-size of f is the minimum size of a circuit for f

main question: what are the complexities of f?

simulations

first simulations

theorem: formulas $\leq$ ABPs $\leq$ circuits

formula of size s

$\rightarrow$ ABP with $w, \ell \leq O(s)$

$\rightarrow$ circuit of size $O(w^3 \cdot \ell)$

next simulations

useful ideas

homogeneous devices

partial derivatives

homogeneous circuits

a circuit is *homogeneous* if all subcomputations are homogeneous

syntactic degrees = semantic degrees

[Strassen]

circuit of size s for f of degree r
$\rightarrow$ homogeneous circuit of size $O(sr^2)$

[Raz]

formula of size s for f of degree r
$\rightarrow$ homogeneous formula of size $poly(s)\binom{r+O(\log s)}{r}$

grading polynomials by degree is very useful

partial derivatives

given a homogeneous circuit for f of degree r

definition $(\partial_v f)$

let v be so that $deg(v) > r/2$

substitute a new variable y instead of v

the new output is $y \cdot \boldsymbol{\partial_v f} + g$

properties

$\star$ $\partial_v f$ and g can be computed by "sub-circuits"

$\star$ $f = f_v \cdot \partial_v f + g$

$\star$ $deg(\partial_v f) = r - deg(v)$

circuits $\rightarrow$ formulas

Hyafil's simulation

theorem [Hyafil 79]

circuit of size s and degree r
$\rightarrow$ formula of depth $O(\log(sr)\log(r))$

$poly(n)$ size and degree $\rightarrow$ $quasi\text{-}poly(n)$ size

Hyafil's simulation

circuit of size s and degree r
$\rightarrow$ formula of depth $O(\log(sr)\log(r))$

sketch

1. w.l.o.g. circuit is homogeneous

2. induction:

find $v = v_1 \times v_2$ so that $deg(v_1) \leq deg(v_2) \leq r/2 < deg(v)$

write

$$f = \partial_v f \cdot f_v + g = \partial_v f \cdot f_{v_1} \cdot f_{v_2} + g$$

degrees of $\partial_v f, f_{v_1}, f_{v_2}$ at most $r/2$

g has smaller circuit

circuits $\rightarrow$ depth 4 formulas

depth 4 simulations

[Agrawal-Vinay 08, Koiran 12, Tavenas 14][1]

a circuit of size s and degree r can be simulated by a homogenous circuit of depth 4 and size

$$2^{O(\sqrt{r \log(sr) \log(n)})}$$

depth 4 is of form $\sum \prod \sum \prod$ with unbounded fanin

[1]chasm (noun):
i. a deep fissure in the earth, rock, or another surface
ii. a profound difference between people, viewpoints, feelings, etc.

depth 4 simulations

saw: circuit of size s and degree r
$\rightarrow$ formula of product depth $O(\log(r))$

depth 4 simulations

saw: circuit of size s and degree r
$\rightarrow$ formula of product depth $O(\log(r))$

idea: cut in middle $\rightarrow$ two depth 2 circuits $\rightarrow$ depth 4

depth 4 simulations

saw: circuit of size s and degree r
$\to$ formula of product depth $O(\log(r))$

idea: cut in middle $\to$ two depth 2 circuits $\to$ depth 4

let V be set of $v = v_1 \times v_2$ with $deg(v) > t$ and $deg(v_i) \leq t$

each f_{v_i} has degree at most t

if we replace each v by a new variable y_v then new output has degree $< r/t$ in Y

depth 4 simulations

circuit of size s and degree $r \to$ formula $\to$ depth 4

let V be set of $v = v_1 \times v_2$ with $deg(v) > t$ and $deg(v_i) \leq t$

"upper part" has $\leq 2^{O(\log(sr)\log(r))}$ variables & degree $< r/t$

"lower parts" have n variables & degree $\leq t$

$\to$ depth 4 circuit of size

$$2^{O(\log(sr)\log(r))\cdot r/t} + 2^{O(\log(sr)\log(r))+t\log(n)}$$
$$= 2^{O(\sqrt{r\log(sr)\log(r)\log(n)})}$$

(worse than Tavenas (VSBR...))

algebraic $\mathsf{P} = \mathsf{NC}_2$

depth $\log^2(\cdot)$ simulations

[Valiant-Skyum-Berkowitz-Rackoff 83]

a circuit of size s and degree r
$\rightarrow$ circuit of size $O(s^3 r^6)$ and depth $O(\log(sr)\log(r))$

depth $\log^2(\cdot)$ simulations

sketch (induction - over simplification)

homoegeneous circuit of size s and degree r

for each v recursively compute f_v by

$$f_v = \sum_u f_u \cdot \partial_u f_v$$

over u so that $deg(u) \approx deg(v)/2$

depth $\log^2(\cdot)$ simulations

sketch (induction - over simplification)

homoegeneous circuit of size s and degree r

for each v recursively compute f_v by

$$f_v = \sum_u f_u \cdot \partial_u f_v$$

over u so that $deg(u) \approx deg(v)/2$

"chain rule"

$$\partial_u f_v = \sum_w \partial_u f_w \cdot \partial_w f_v$$

circuits $\rightarrow$ depth 3 circuits

depth 3 simulations

[Gupta-Kamath-Kayal-Saptharishi 13]

over $\mathbb{Q}$ a circuit of size s and degree r can be simulated by circuit of depth 3 and size

$$2^{O(\sqrt{r \log(sr) \log(n)})}$$

comments:

(i) non homogeneous: degree $\approx$ size

(ii) over fields of large characteristic (necessary)

depth 3 simulations

overview (quantitatively inaccurate)

circuit of size s and degree r

$\rightarrow \sum\prod^{(\sqrt{r})}\sum\prod^{(\sqrt{r})}$ circuit of size $2^{\sqrt{r}}$

$\rightarrow \sum\bigwedge^{(\sqrt{r})}\sum\sum\bigwedge^{(\sqrt{r})}\sum$ circuit of size $2^{\sqrt{r}}$

$\rightarrow \sum\prod\sum$ circuit of size $2^{\sqrt{r}}$

$\bigwedge$ is a powering gate

$\sum\prod\sum\prod \to \sum\bigwedge\sum\bigwedge\sum$

[Fischer-Ryser]

$\prod^{(d)} \to \sum^{(2^d)} \bigwedge^{(d)} \sum$:

$$\prod_{i=1}^{d} x_i = \frac{(-1)^d}{d!} \sum_{T \subseteq [d]} (-1)^{|T|} \left(\sum_{i \in T} x_i \right)^d$$

$\sum\prod\sum\prod \to \sum\bigwedge\sum\bigwedge\sum$

[Fischer-Ryser]

$\prod^{(d)} \to \sum^{(2^d)} \bigwedge^{(d)} \sum$:

$$\prod_{i=1}^{d} x_i = \frac{(-1)^d}{d!} \sum_{T\subseteq[d]} (-1)^{|T|} \left(\sum_{i\in T} x_i\right)^d$$

apply to two $\sum\prod$ circuits and merge

$$\sum \prod^{(\sqrt{r})} \sum \prod^{(\sqrt{r})} \to \sum \bigwedge^{(\sqrt{r})} \sum \bigwedge^{(\sqrt{r})} \sum$$

$\sum \bigwedge \sum \bigwedge \sum \to \sum \prod \sum$

duality trick [Saxena, Shpilka-Wigderson]
$\bigwedge \sum \to *$: there are uni-variate polynomial g_{ij} so that

$$(x_1 + \ldots + x_m)^d = \sum_{i=1}^{md+1} \prod_{j=1}^{m} g_{ij}(x_j)$$

$\sum \bigwedge \sum \bigwedge \sum \to \sum \prod \sum$

duality trick [Saxena, Shpilka-Wigderson]
$\bigwedge \sum \to *$: there are uni-variate polynomial g_{ij} so that

$$(x_1 + \ldots + x_m)^d = \sum_{i=1}^{md+1} \prod_{j=1}^{m} g_{ij}(x_j)$$

to a $\sum \prod \sum$ circuit:

$$\sum \bigwedge \sum \bigwedge \sum = \sum_t \sum_i \prod_j g_{ij}(\ell_{tij}^d) \qquad \text{(apply to left } \bigwedge\text{)}$$

$$= \sum_t \sum_i \prod_\ell (\ell_{ti\ell} - \alpha_{ti\ell}) \qquad \text{(factor over } \mathbb{C}\text{)}$$

(to go from $\mathbb{C}$ to $\mathbb{Q}$ need to pay some more) summary

several depth reductions

useful for

- computations
- lower bounds

importance of homogeneous polynomials

importance of grading (degree, ...)