Lecture 2: Algebraic Models of Computation

Rafael Oliveira

University of Waterloo
Cheriton School of Computer Science

rafael.oliveira.teaching@gmail.com

January 12, 2021

Overview

- Algebraic Models of Computation
- Operations in Algebraic Circuits
- Conclusion
- Acknowledgements

Dense Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Dense representation*: $p(x_1, \ldots, x_n)$ of degree d in $R[x_1, \ldots, x_n]$ is represented as a list of *all monomials of degree* $\leq d$ and *their coefficients* in p.

Dense Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Dense representation*: $p(x_1, \ldots, x_n)$ of degree d in $R[x_1, \ldots, x_n]$ is represented as a list of *all monomials of degree $\leq d$* and *their coefficients* in p.
- Examples:
  - $p(x, y) = \underline{xy}$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

$$p(x, y) \rightarrow [\underset{\substack{\uparrow \\ \text{degree}}}{2}, (0, x^2), (1, xy), (0, y^2), (0, x), (0, y), (0, 1)]$$

$$p(x,y) = 0 \cdot x^2 + 1 \cdot xy + 0 \cdot y^2 + 0 \cdot x + 0 \cdot y + 0 \cdot 1$$

Dense Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Dense representation*: $p(x_1, \ldots, x_n)$ of degree d in $R[x_1, \ldots, x_n]$ is represented as a list of *all monomials of degree $\leq d$* and *their coefficients* in p.
- Examples:
  - $p(x, y) = xy$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

$$p(x, y) \to [2, (0, x^2), (1, xy), (0, y^2), (0, x), (0, y), (0, 1)]$$

  - $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

$$q(x, y) \to [\underline{2,} (0, x^2), \underline{(1, xy)}, (0, y^2), \underline{(-3, x)}, (0, y), \underline{(1, 1)}]$$

Dense Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Dense representation*: $p(x_1, \ldots, x_n)$ of degree d in $R[x_1, \ldots, x_n]$ is represented as a list of *all monomials of degree $\leq d$* and *their coefficients* in p.
- Examples:
  - $p(x, y) = xy$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

    $$p(x, y) \rightarrow [2, (0, x^2), (1, xy), (0, y^2), (0, x), (0, y), (0, 1)]$$
  - $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

    $$q(x, y) \rightarrow [2, (0, x^2), (1, xy), (0, y^2), (-3, x), (0, y), (1, 1)]$$

- Very wasteful for multivariate polynomials, or polynomials with high degree. Needs to store all $\binom{n+d}{d}$ coefficients!

monomials n vars
deg $\leq d$.

Dense Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Dense representation*: $p(x_1, \ldots, x_n)$ of degree d in $R[x_1, \ldots, x_n]$ is represented as a list of *all monomials of degree $\leq d$* and *their coefficients* in p.
- Examples:
  - $p(x, y) = xy$ polynomial of degree 2 over $\mathbb{Q}[x, y]$
  $$p(x, y) \to [2, (0, x^2), (1, xy), (0, y^2), (0, x), (0, y), (0, 1)]$$
  - $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$
  $$q(x, y) \to [2, (0, x^2), (1, xy), (0, y^2), (-3, x), (0, y), (1, 1)]$$
  $(0,(2,0)),(1,(1,1))$
- Very wasteful for multivariate polynomials, or polynomials with high degree. Needs to store all $\binom{n+d}{d}$ coefficients!
- In this class, we will represent a monomial $x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n}$ either by writing the monomial explicitly, or by its *exponent vector*
$$\underline{x_1^{e_1} x_2^{e_2}} \cdots x_n^{e_n} \leftrightarrow \underline{(e_1, \ldots, e_n)}$$

Sparse Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Sparse representation*: $p(x_1, \ldots, x_n)$ in $R[x_1, \ldots, x_n]$ is represented as a list of all *non-zero* monomials and *their coefficients* in p.

Sparse Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Sparse representation*: $p(x_1, \ldots, x_n)$ in $R[x_1, \ldots, x_n]$ is represented as a list of all *non-zero* monomials and *their coefficients* in p.
- Examples:
  1. $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

$$q(x, y) \to [(1, xy), (-3, x), (1, 1)]$$

Sparse Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Sparse representation*: $p(x_1, \ldots, x_n)$ in $R[x_1, \ldots, x_n]$ is represented as a list of all *non-zero* monomials and *their coefficients* in p.
- Examples:
  1. $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

$$q(x, y) \to [(1, xy), (-3, x), (1, 1)]$$

  2. $p(x_1, \ldots, x_n) = \prod_{i=1}^{n} (x_i + 1)$ $\leftarrow$ 2^n entries

Too many coefficients even for some "simple polynomials."

$S \subset [n] := \{1, 2, \ldots, n\}$ $\quad 2^n$

$\left(1, x_S := \prod_{i \in S} x_i\right)$

Sparse Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Sparse representation*: $p(x_1, \ldots, x_n)$ in $R[x_1, \ldots, x_n]$ is represented as a list of all *non-zero* monomials and *their coefficients* in p.
- Examples:
  1. $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

     $$q(x, y) \to [(1, xy), (-3, x), (1, 1)]$$

  2. $$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} (x_i + 1)$$

     Too many coefficients even for some "simple polynomials."

  3. $$\mathrm{Det}(X) = \sum_{\sigma \in S_n} (-1)^{\sigma} \prod_{i \in [n]} X_{i\sigma(i)}$$ $n!$

     Too many coefficients too, and determinant also "simple polynomial."

Sparse Representation

- Setting: polynomial ring $R[x_1, \ldots, x_n]$
- *Sparse representation*: $p(x_1, \ldots, x_n)$ in $R[x_1, \ldots, x_n]$ is represented as a list of all *non-zero* monomials and *their coefficients* in p.
- Examples:
  1. $q(x, y) = xy - 3x + 1$ polynomial of degree 2 over $\mathbb{Q}[x, y]$

     $$q(x, y) \to [(1, xy), (-3, x), (1, 1)]$$

  2. $p(x_1, \ldots, x_n) = \prod_{i=1}^{n} (x_i + 1)$

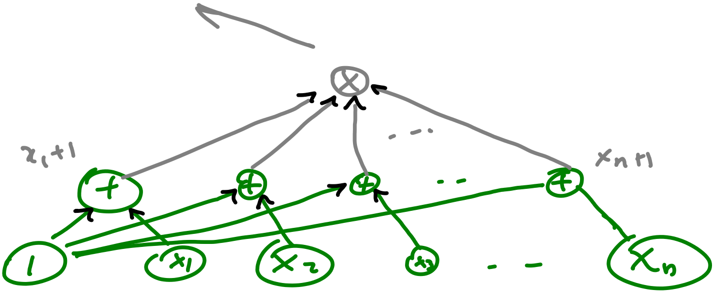
     Too many coefficients even for some "simple polynomials."
  3. $\mathrm{Det}(X) = \sum_{\sigma \in S_n} (-1)^{\sigma} \prod_{i \in [n]} X_{i\sigma(i)}$

     Too many coefficients too, and determinant also "simple polynomial."
- Why do we think that the polynomials from examples # 2 & 3 are "simple?"

Algebraic Circuits - base ring R

- Models the *amount of operations* needed to compute polynomial

$$\prod (x_i + 1)$$

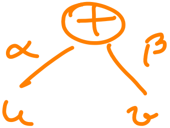


Algebraic Circuits - base ring R

- Models the *amount of operations* needed to compute polynomial
- *Algebraic Circuit*: directed acyclic graph Φ with
  - input gates labelled by variables $x_1, \ldots, x_n$ or elements of R

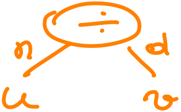
Algebraic Circuits - base ring R

- Models the *amount of operations* needed to compute polynomial
- *Algebraic Circuit*: directed acyclic graph Φ with
  - input gates labelled by variables $x_1, \ldots, x_n$ or elements of R
  - other gates labelled $+, \times, \div$
  - $\div$ gate takes two inputs, which are labelled numerator/denominator



$\alpha u + \beta v$

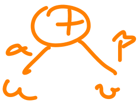
$\alpha, \beta \in R$



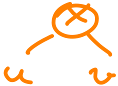
$\frac{u}{v}$

Algebraic Circuits - base ring R

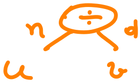
- Models the *amount of operations* needed to compute polynomial
- *Algebraic Circuit*: directed acyclic graph Φ with
  - input gates labelled by variables $x_1, \ldots, x_n$ or elements of R
  - other gates labelled $+, \times, \div$
  - $\div$ gate takes two inputs, which are labelled numerator/denominator
  - gates compute polynomial (rational function) in natural way



$\alpha u + \beta v$



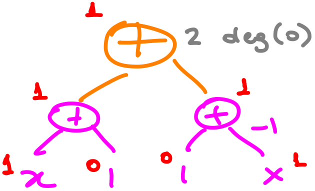
uv

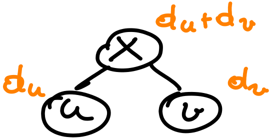


$\frac{u}{v}$

Algebraic Circuits - base ring R

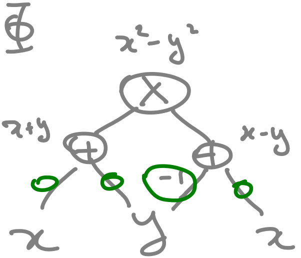
- Models the *amount of operations* needed to compute polynomial
- *Algebraic Circuit*: directed acyclic graph Φ with
  - input gates labelled by variables $x_1, \ldots, x_n$ or elements of R
  - other gates labelled $+, \times, \div$
  - $\div$ gate takes two inputs, which are labelled numerator/denominator
  - gates compute polynomial (rational function) in natural way
- **formal degree of a gate:** the degree of a gate is defined inductively
  - if input gate: degree is 0 if gate is element of the field, 1 if it is a variable
  - $u = w + v$ then $\deg(u) = \max(\deg(w), \deg(v))$
  - $u = w \times v$ then $\deg(u) = \deg(w) + \deg(v)$





Complexity Measures in Algebraic Circuits

- *circuit size:* number of edges in the circuit, denoted by $\mathcal{S}(\Phi)$
- *cost of ring elements:* in classical algebraic complexity, there is unit cost for the use of any base ring element
- Sometimes we will add bit complexity of base ring elements



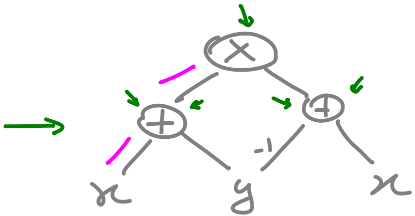
$\mathcal{S}(\Phi) = 6$

Complexity Measures in Algebraic Circuits

amount of operations

- *circuit size:* number of edges in the circuit, denoted by $\mathcal{S}(\Phi)$
- *cost of ring elements:* in classical algebraic complexity, there is unit cost for the use of any base ring element
- Sometimes we will add bit complexity of base ring elements
- *circuit depth:* length of longest direct path from an input to an output

parallel complexity of problem

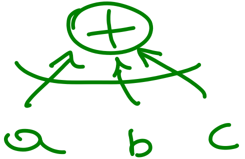


$\text{depth}(\Phi) = 2.$

Complexity Measures in Algebraic Circuits

- *circuit size:* number of edges in the circuit, denoted by $\mathcal{S}(\Phi)$
- *cost of ring elements:* in classical algebraic complexity, there is unit cost for the use of any base ring element
- Sometimes we will add bit complexity of base ring elements
- *circuit depth:* length of longest direct path from an input to an output
- *constant depth circuits:* for circuits of constant depth, we don't place restriction on the fan-in of an ~~edge~~ gate.

indegree



3

if general circuits assume fan-in ≤ 2

Examples - Constant Depth Circuits

$\boxed{\Sigma \Pi} \leftarrow$ depth 2 circuit

$$\underbrace{\sum \underbrace{\alpha_{\bar{e}}}_{\in R} \cdot \underbrace{x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n}}_{\text{monomial}}}_{\alpha_{\bar{e}} \cdot \prod x_i^{e_i}}$$

$\boxed{\prod (x_i + 1)}$

$\boxed{\cancel{\Pi \Sigma}}$

$\Sigma \Pi \underbrace{\Sigma}_{\text{linear forms}}$

$$\sum_{i=1}^{s} \prod_{j=1}^{d} \underbrace{l_{ij}(x_1, \ldots, x_n)}_{(x_i + 1)}$$

$s = 1$

$\Sigma \Pi \overbrace{\Sigma \Pi}^{\text{sparse polynomials}}$

Obtaining Homogeneous Components

Theorem ([Strassen 1973])

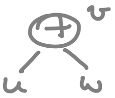
If a polynomial $p(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ can be computed by a circuit Φ of size $\mathcal{S}(\Phi)$, then the homogeneous components $H_0[p], H_1[p], \ldots, H_r[p]$ can be computed by a homogeneous *circuit of size $O(r^2 \cdot \mathcal{S}(\Phi))$.*

Proof: induction on depth

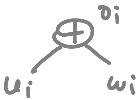
input are homogeneous ✓

for every gate v $\quad v_0, v_1, \ldots, v_r$

$v_i \leftarrow$ ith homogeneous component of P_v



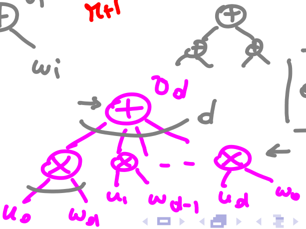
$v_i = u_i + w_i$



$r+1$

$$v_d = \sum_{i=0}^{d} u_i \cdot w_{d-i}$$

$$O(0+1+\cdots+r) = O(r^2)$$



add $O(d)$ gates

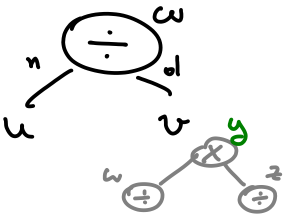
Obtaining Homogeneous Components

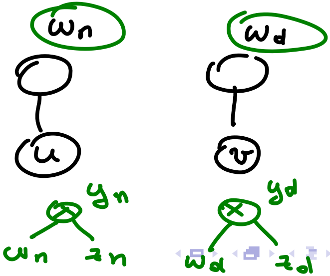
Getting rid of Division

Theorem ([Strassen 1973])

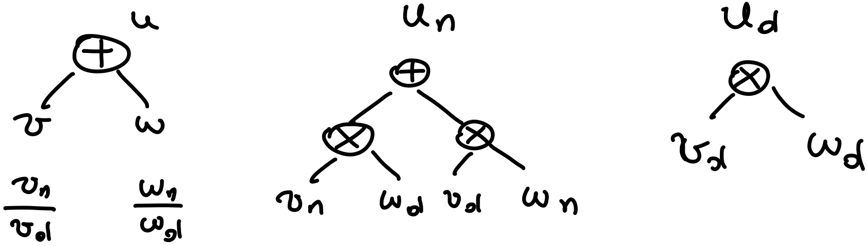
If a polynomial $p(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ *of degree* d *can be computed by a circuit* Φ *of size* $\mathcal{S}(\Phi)$ *using* $+, \times, \div$, *then there is a circuit* Ψ *of size* $\mathrm{poly}(\mathcal{S}(\Phi), d, n)$ *which computes* p *without using division gates.*

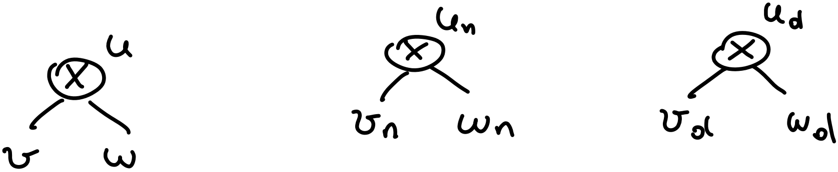
Proof: show that Φ can be made have only one division gate, on the output gate





Getting rid of Division

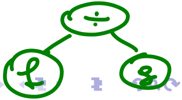




size new circuit $O(s)$

P output $\frac{f}{g}$ ← numerator of output
←

P



Getting rid of Division

$$P = \frac{f}{g} \qquad g = \underset{\uparrow}{g_0} - \hat{g} \leftarrow \text{sum terms deg} \geq 1$$

constant term

deg(p) = d

$$g_0 = 1$$

$$\frac{1}{g} = \frac{1}{1-\hat{g}} = 1 + \underbrace{\hat{g}}_{\geq 1} + \underbrace{\hat{g}^2}_{\geq 2} + \cdots + \underbrace{\hat{g}^d}_{\geq d} + \underbrace{\underbrace{\hat{g}^{d+1}}_{\geq d+1} + \cdots}_{\text{do NOT contribute computation of } P}$$

$$P = \underbrace{H_{\leq d}}_{\text{small ckt.}} \underbrace{\left[\underbrace{f}_{\text{small ckt}} \underbrace{\left(1 + \hat{g} + \hat{g}^2 + \cdots + \hat{g}^d\right)}_{\text{small ckt}} \right]}_{\text{poly with small ckt no divisions}}$$

small ckt

Computing Determinants with Small Circuits

Corollary

The polynomial Det(X) *can be computed by an arithmetic circuit of* poly(n) *size.*

$$ad - bc$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \longrightarrow \begin{pmatrix} a & b \\ 0 & d - \frac{bc}{a} \end{pmatrix}$$

$$a \cdot \left(d - \frac{bc}{a}\right) = ad - bc$$

Gaussian Elimination gives ckt computes Det(X) using divisions $\underline{O(n^3)}$ operations

$\Longrightarrow$ by S'73 ckt without division.

Computing Determinants with Small Circuits

- Algebraic Models of Computation
- Operations in Algebraic Circuits
- Conclusion
- Acknowledgements

Conclusion

In today's lecture, we learned about different computational models for symbolic computation, and basic computations in these models.

- Dense representation
- Sparse representation
- Algebraic circuits
- Proved that the determinant can be computed by algebraic circuits of polynomial size

Acknowledgement

- Algebraic circuit part of lecture largely based on chapters 1 & 2 of survey

  https://www.nowpublishers.com/article/Details/TCS-039

References I

Strassen, Volker 1973.
Vermeidung von Divisionen
The Journal fur die Reine und Angewandte Mathematik