- Let $X = (x_{ij})_{n\times n}$ be the matrix whose $\det_n$ we want to compute as a "small" arithmetic circuit.

- Gaussian elimination uses division, permutation, etc. that we cannot naively write as a polynomial.

  So, we will use a different idea – Newton's identity. This is based on Leverrier's method (1840) & was used by Csanky (1976).

- Idea: • $\det(X) = \prod_{i\in[n]} \lambda_i$ where λ_i's are the eigenvalues of X, in $\overline{\mathbb{Q}(\overline{x})}$.

  • The plan is to express $\prod \lambda_i$ as a polynomial in the power sums :
  $$p_k := \sum_{i\in[n]} \lambda_i^k \quad , \text{ for } k \geq 0 .$$

  • Note that $tr(X^k) = p_k$, for $k \geq 0$. Thus, the above promised expression would give us a $O(\lg^2 n)$-depth, poly(n)-size, circuit for det(X). (Divide & conquer?)

• The power-sum formulas are obtained by studying the elementary symmetric polynomials:

$$e_k(\lambda_1,\ldots,\lambda_n) := \sum_{S\in\binom{[n]}{k}} \lambda_S \quad , \text{ for } k \geq 0.$$

Eg. $e_0(\bar\lambda) := 1$, $e_1 = \lambda_1 + \cdots + \lambda_n$, $\ldots$, $e_n = \lambda_1 \cdots \lambda_n$,
$e_{n+1} = 0, \ldots$

<u>Lemma</u> (Newton's identity): For $k, n \geq 1$,

$$k \cdot e_k(\bar\lambda) = \sum_{i\in[k]} (-1)^{i-1} \cdot e_{k-i}(\bar\lambda) \cdot p_i(\bar\lambda) .$$

[Eg. $e_1 = p_1$, $2e_2 = e_1 p_1 - p_2$, $3e_3 = e_2 p_1 - e_1 p_2 + p_3$.]
[In fact, $\mathbb{Q}[e_1,\ldots,e_n] = \mathbb{Q}[p_1,\ldots,p_n]$.]

<u>Proof</u>:

• Let us consider a generating function & a <u>formal power series</u> viewpoint:

$$\sum_{k=0}^{n} e_k \cdot (-t)^k = \prod_{i\in[n]} (1-\lambda_i t).$$

- Apply $\underline{t\cdot\partial_t}$ both sides (i.e. differentiate & scale-up)

$$\sum_{0\le k\le n} k e_k \cdot (-t)^k = t\cdot \sum_{i\in[n]} (-\lambda_i)\cdot \prod_{j\neq i} (1-\lambda_j t)$$

$$= -\left(\sum_{i\in[n]} \frac{\lambda_i t}{1-\lambda_i t}\right)\cdot \prod_{j\in[n]} (1-\lambda_j t)$$

$$= -\left(\sum_{i\in[n]} \sum_{j\ge 1} (\lambda_i t)^j\right)\cdot \prod_j (1-\lambda_j t)$$

$$= \left(\sum_{j\ge 1} p_j t^j\right)\cdot\left(\sum_{i\in[n]} (-1)^{i-1}\cdot e_i t^i\right)$$

$$= \sum_{j,k\ge 1} p_j t^j \cdot (-1)^{k-j-1}\cdot e_{k-j}\cdot t^{k-j}$$

$$= \sum_{k\ge 1} (-t)^k \cdot\left(\sum_{j\ge 1} (-1)^{j-1}\cdot e_{k-j}\cdot p_j\right).$$

$\square$

- The <u>triangular-matrix</u> structure of this recurrence can be utilized to construct a $O(\lg^2 n)$-depth, poly(n)-size circuit for $e_n = \det(X)$ over $\mathbb{Q}$.

The same method was extended by Schönhage (1993) to positive characteristic fields.

<u>Theorem</u>: $\det_n \in VP$ (depth-$\lg^2 n$) (P-uniform).

(bounded fanin/fanout)

<u>Pf</u>:

- size & depth analysis is left as an exercise.
- The key ideas are to compute n powers of a matrix & solve a triangular system in the lowest depth possible.

$\square$

- $\det_n$ is closely related to another important polynomial representation— <u>Arithmetic branching program</u> (<u>ABP</u>).