My lecture at the Conference on Algebraic Complexity in the memory of Jacques Morgenstern, INRIA Sophia Antipolis, France May 1995.

Complexity Theory in the Service of Algorithm Design

Erich Kaltofen

Rensselaer

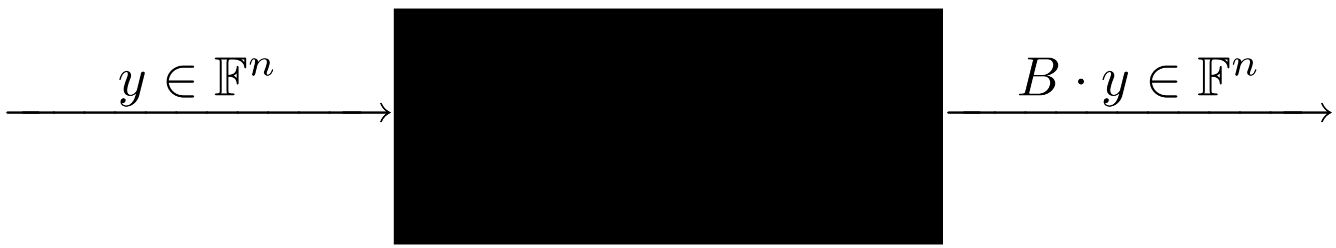
Rensselaer Polytechnic Institute
Department of Computer Science
Troy, New York, USA

Outline

- **Wiedemann's sparse linear system solver**
  - Coordinate recurrences
  - More applications of the transposition principle
- **Reverse mode of automatic differentiation**
  - Transposition principle by derivatives
  - More applications
- **Polynomial factorization**
  - Berlekamp's polynomial factorization algorithm
  - use of the Wiedemann method
  - new baby step/giant step algorithm

A “black box” matrix

is an efficient **procedure** with the specifications



$B \in \mathbb{F}^{n \times n}$

$\mathbb{F}$ an arbitrary field

i.e., the matrix is not stored explicitly, its structure is unknown.

Main algorithmic problem: How to efficiently solve a linear system with a black box coefficient matrix?

Idea for Wiedemann's algorithm

$B \in \mathbb{F}^{n\times n}$, $\mathbb{F}$ a (possibly finite) field

$\phi^B(\lambda) = c'_0 + c'_1\lambda + \cdots + c'_m\lambda^m \in \mathbb{F}[\lambda]$ **minimum polynomial** of B:

$\forall\, u, v \in \mathbb{F}^n$: $\forall\, j \geq 0$: $u^{\mathrm{tr}} B^j \phi^B(B) v = 0$

$$\Updownarrow$$

$$c'_0 \cdot \underbrace{u^{\mathrm{tr}} B^j v}_{a_j} + c'_1 \cdot \underbrace{u^{\mathrm{tr}} B^{j+1} v}_{a_{j+1}} + \cdots + c'_m \cdot \underbrace{u^{\mathrm{tr}} B^{j+m} v}_{a_{j+m}} = 0$$

$$\Updownarrow$$

$\{a_0, a_1, a_2, \ldots\}$ is generated by a linear recursion

Theorem (Wiedemann 1986): *For random* $u, v \in \mathbb{F}^n$, *a linear generator for* $\{a_0, a_1, a_2, \ldots\}$ *is one for* $\{I, B, B^2, \ldots\}$.

$$\forall\, j \geq 0\colon c_0 a_j + c_1 a_{j+1} + \cdots + c_d a_{j+d} = 0$$

$$\Big\Downarrow \text{(with high probability)}$$

$$c_0 B^j v + c_1 B^{j+1} v + \cdots + c_d B^{j+d} v = 0$$

$$\Big\Downarrow \text{(with high probability)}$$

$$c_0 B^j + c_1 B^{j+1} + \cdots + c_d B^{j+d} = 0$$

that is, $\phi^B(\lambda)$ divides $c_0 + c_1\lambda + \cdots + c_m\lambda^m$

Algorithm *Homogeneous Wiedemann*

Input: $B \in \mathbb{F}^{n\times n}$ singular

Output: $w \neq \mathbf{0}$ such that $Bw = \mathbf{0}$

Step W1: Pick random $u, v \in \mathbb{F}^n$; $\quad b \leftarrow Bv$;
for $i \leftarrow 0$ **to** $2n-1$ **do** $a_i \leftarrow u^{\mathrm{tr}} B^i b$.
(Requires $2n$ black box calls.)

Step W2: Compute a linear recurrence generator for $\{a_i\}$,
$c_\ell \lambda^\ell + c_{\ell+1}\lambda^{\ell+1} + \cdots + c_d \lambda^d, \quad \ell \geq 0, d \leq n, c_\ell \neq 0,$
by the Berlekamp/Massey algorithm.

Step W3: $\widehat{w} \leftarrow c_\ell v + c_{\ell+1} Bv + \cdots + c_d B^{d-\ell} v$;
(With high probability $\widehat{w} \neq 0$ and $B^{\ell+1}\widehat{w} = 0$.)
Compute first k with $B^k \widehat{w} = 0$; **return** $w \leftarrow B^{k-1}\widehat{w}$.

Steps W1 and W3 have the same computational complexity

$$u^{\mathrm{tr}} \cdot [\, v \mid Bv \mid B^2 v \mid \ldots \mid B^{2n} v \,] = [\, a_{-1} \quad a_0 \quad a_1 \quad \ldots \quad a_{2n-1} \,]$$

$$[\, v \mid Bv \mid B^2 v \mid \ldots \mid B^{2n} v \,] \cdot \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{2n} \end{bmatrix} = w$$

Fact: $X \cdot y$ and $X^{\mathrm{tr}} \cdot z$ have the same computational complexity [Kaminski *et al.*, 1988].